

Team Agreement

Team 7

Everyone in the team can propose changes to our working agreements. You can also always suggest new working agreements in the team retrospectives.

Contents

1. Project overview
2. Initial requirements specification by the client
3. Team members and their roles
4. Culture
5. IDE & Tools
6. Meeting timings
7. Gitlab pushing code rules
8. Definition of done

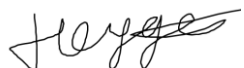

Project overview


- We're building a website to develop and implement a pilot Recruitment and Admissions application for the School of Healthcare Sciences.
-
- **Client:** Janet Israel – Cardiff University
- **Current status (Nov 24th 2024):** The project has just started.

Initial requirements specification

[View client meeting recording](#)

• Team members and their roles

• Person	• Role	Contacts	Agreement
Tomos Leggett	• Developer	leggettjt@cardiff.ac.uk	
Wei Peng	Developer	pengw2@cardiff.ac.uk	

Mariia Moroz	Developer	morozm@cardiff.ac.uk	
Snigdha Banani	Developer	bananis@cardiff.ac.uk	<i>Snigdha Banani</i>
Yangshuo Chen	Developer	cheny247@cardiff.ac.uk	<i>Yangshuo</i>

Culture

As a team, we agree to...

- Respect, listen to, and try to understand each other's opinions
- Ask for help when we need it
- Are trying to be transparent and honest
- Actively give constructive feedback
- Agree to be transparent and honest
- Encourage a diversity of opinions on all topics

IDE & Tools

IDE	IntelliJ Idea
Java Version	17
CSS Framework	Bootstrap

Meeting timings

- As a team, we agree to...
- - Scrum call meet: Daily
 - Planning meet: Friday
 - Demo with client meet after every sprint: Every week/client meeting (Every THU)
 - Retrospective meet: Thursday

Gitlab pushing code rules

As a team, we agree to...

- Never commit directly to the main branch
- Create a new branch for every feature/bugfix

Branches:

- "main" branch: Final merge of working code at the end of each sprint before showing demo
- "development" branch: Every week merging of working code after each feature. Should take pull from this branch only
- "feature/<name>" branch: Should create own respective branch and work in the local
- Code review: Always tag another team member as reviewer while creating a PR. Never merge/ keep yourself as the reviewer
- Merge request: If added as a reviewer, before accepting and merging the code add review points in the user stories with respective commit id
- Flow of pushing code:
 - "feature/<name>" branch -> "development" branch
 - "development" branch -> "main" branch at the final stage if the code is working
- If unsure of anything take help but don't push as it may overwrite other codes

Definition of Ready

As a team, we agree that a task is done, when...

- Peer code review
- No known defects
- All tests have passed
- Code is merged to the main branch and deployed successfully
 - Estimation/sizing

Definition of Done

As a team, we agree that a task is done, when...

- Task acceptance criteria met
- Peer code review passes
- Client gives feedbacks