

# *Git,* Gestion de branches sous IntelliJ

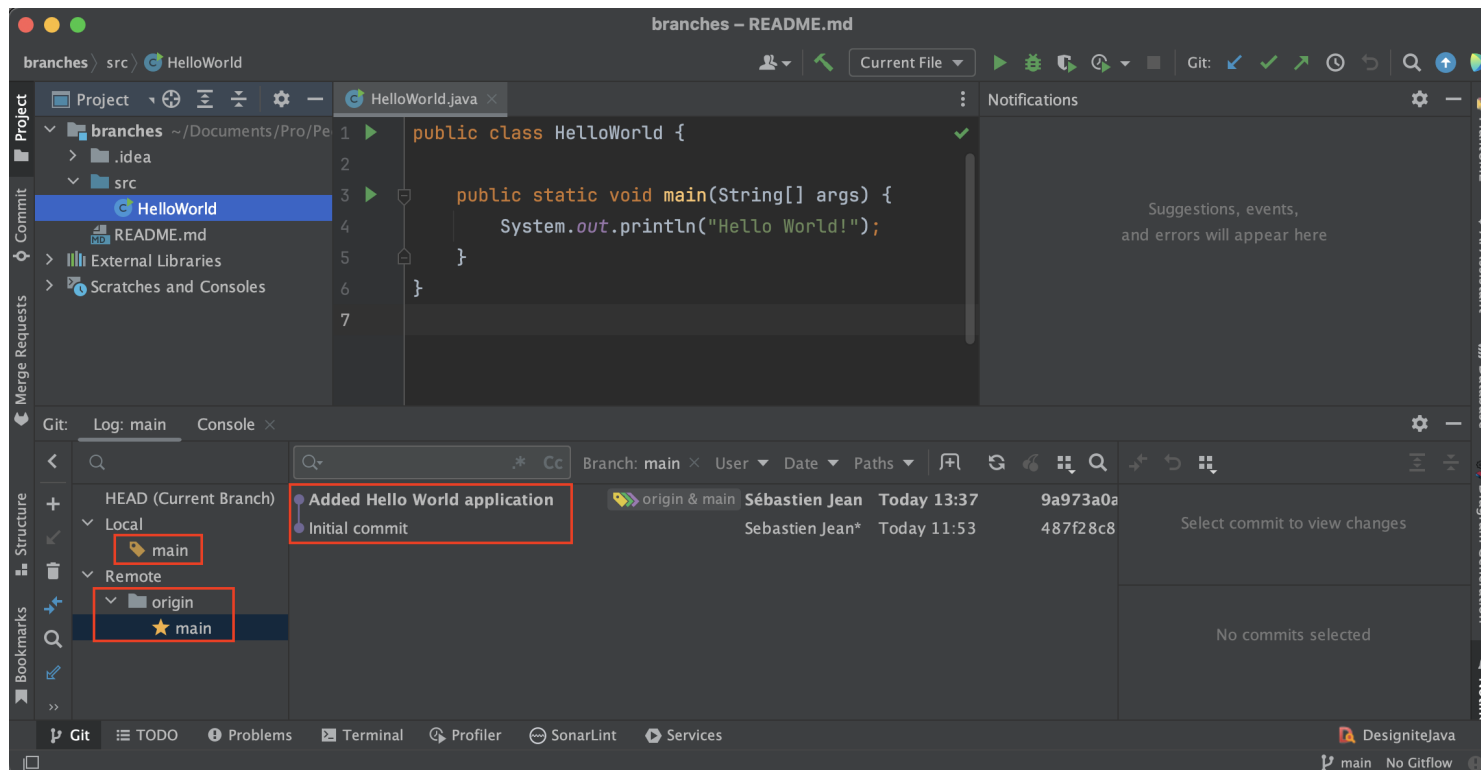
**Sébastien Jean**

IUT de Valence  
Département Informatique

v1.0, 5 décembre 2023

# Point de départ

- **Dépôt local initial** : une branche unique **main** (synchronisée) et **2** commits



# Point de départ

- Dépôt distant initial : une branche unique `main` et 2 commits

The screenshot shows a GitLab repository page for 'branches' (Project ID: 5141). The repository is a dummy repository for git branching training using IntelliJ. It has 2 commits and 1 branch (main). The main branch is selected. The commit history shows two commits: 'Added Hello World application' and 'Initial commit'. The file structure includes .idea, src, and README.md. The README.md file is open, showing the title 'Git-Branches' and the description 'Dummy repository for git branching training using IntelliJ.'

Sebastien Jean > branches

**branches**  
Project ID: 5141

2 Commits 1 Branch 0 Tags 14 KiB Project Storage

Dummy repository for git branching training using IntelliJ

Added Hello World application  
Sébastien Jean authored 1 hour ago

main branches / +

History Find file Edit ↓ ↓ Clone ↓

README Add LICENSE Add CHANGELOG Add CONTRIBUTING Set up CI/CD Configure Integrations

Name	Last commit	Last update
.idea	Initial commit	24 minutes ago
src	Added Hello World application	24 minutes ago
README.md	Initial commit	24 minutes ago

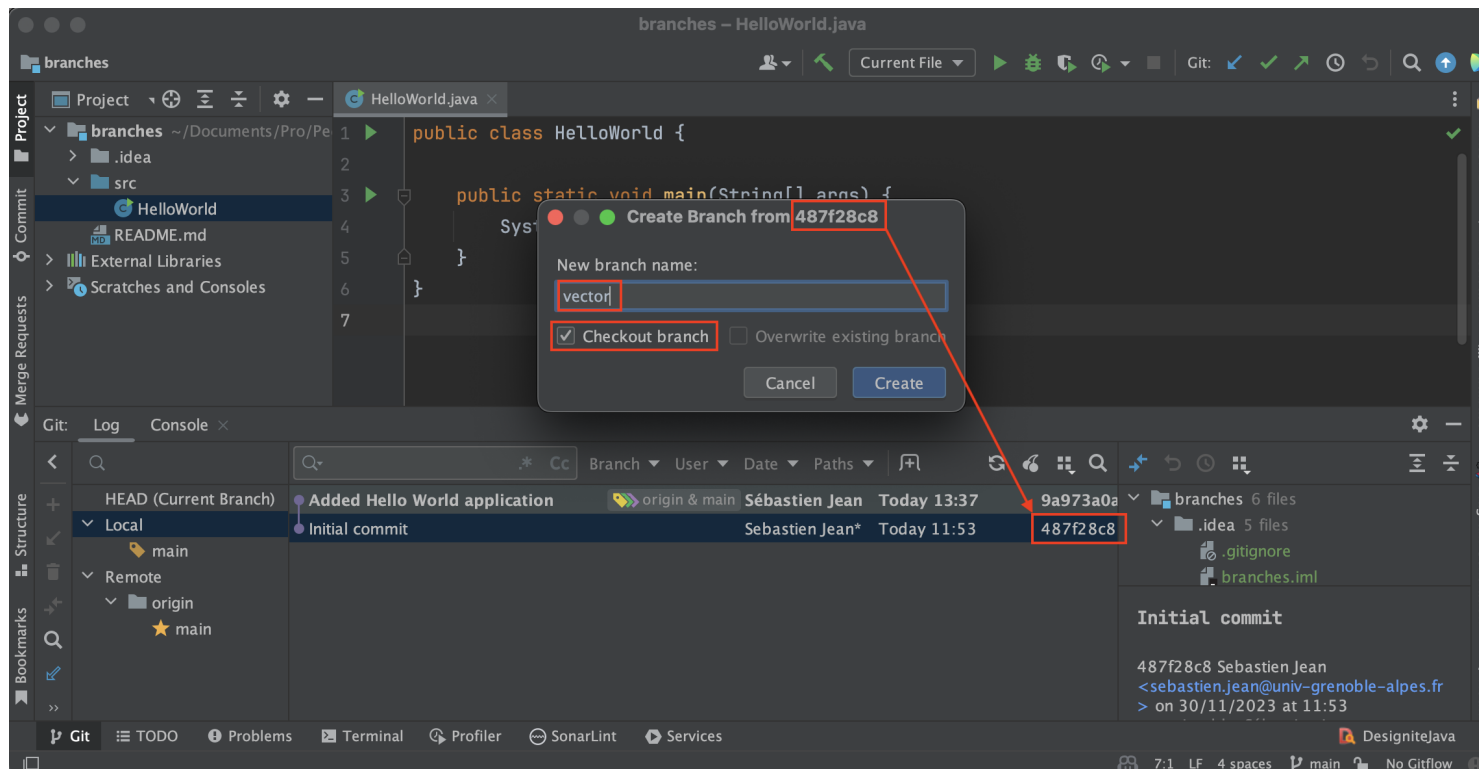
README.md

### Git-Branches

Dummy repository for git branching training using IntelliJ.

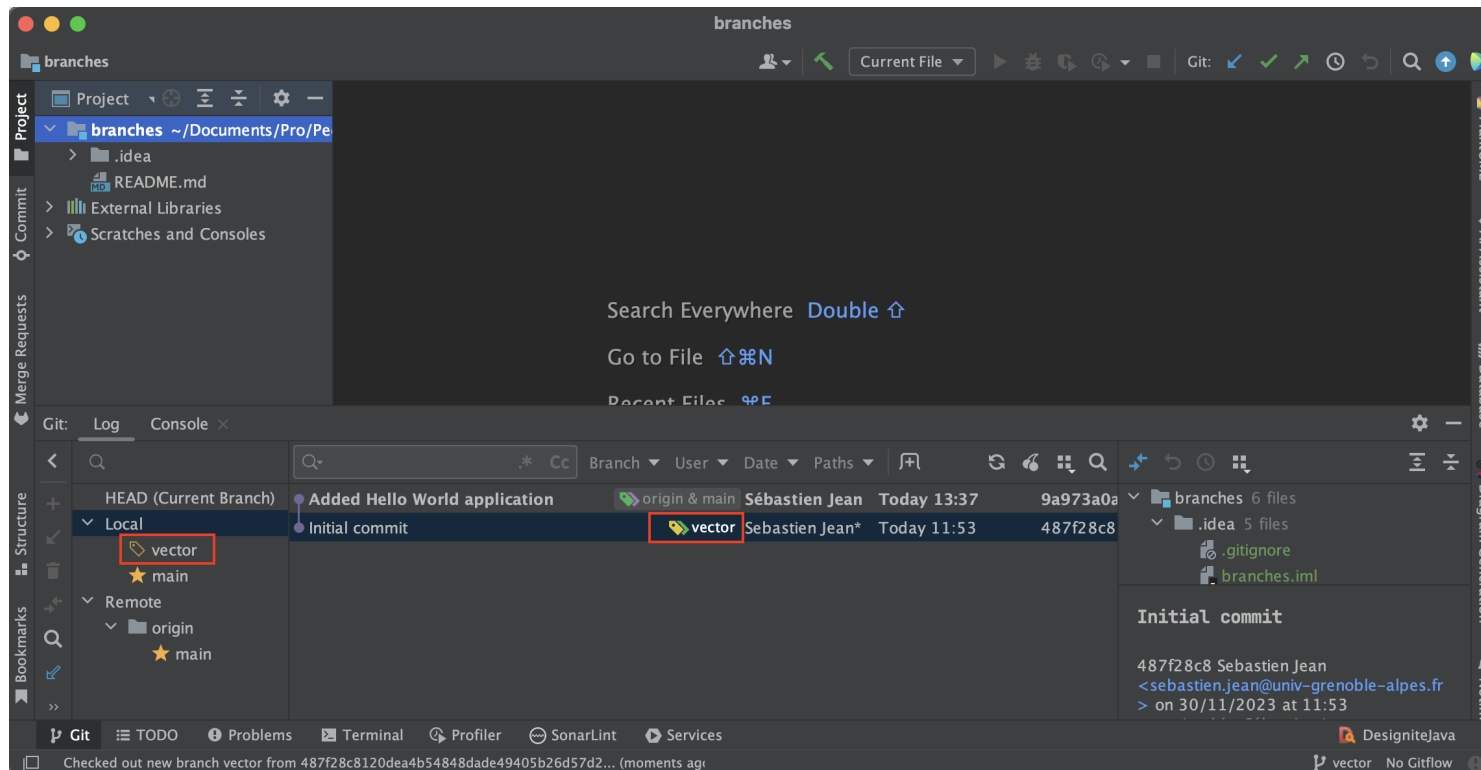
# Créer une nouvelle branche

- En se plaçant sur un **commit** (ici 487f28c8)
- En indiquant un **nom** de branche
- Avec en **option** la possibilité de se **déplacer immédiatement** sur la nouvelle branche



# Créer une nouvelle branche, suite

- La branche est créée **localement**
- Le **répertoire de travail** reflète le **dernier commit** sur la branche
- Avec en **option** la possibilité de se **déplacer immédiatement** sur la nouvelle branche



# Créer une nouvelle branche, suite

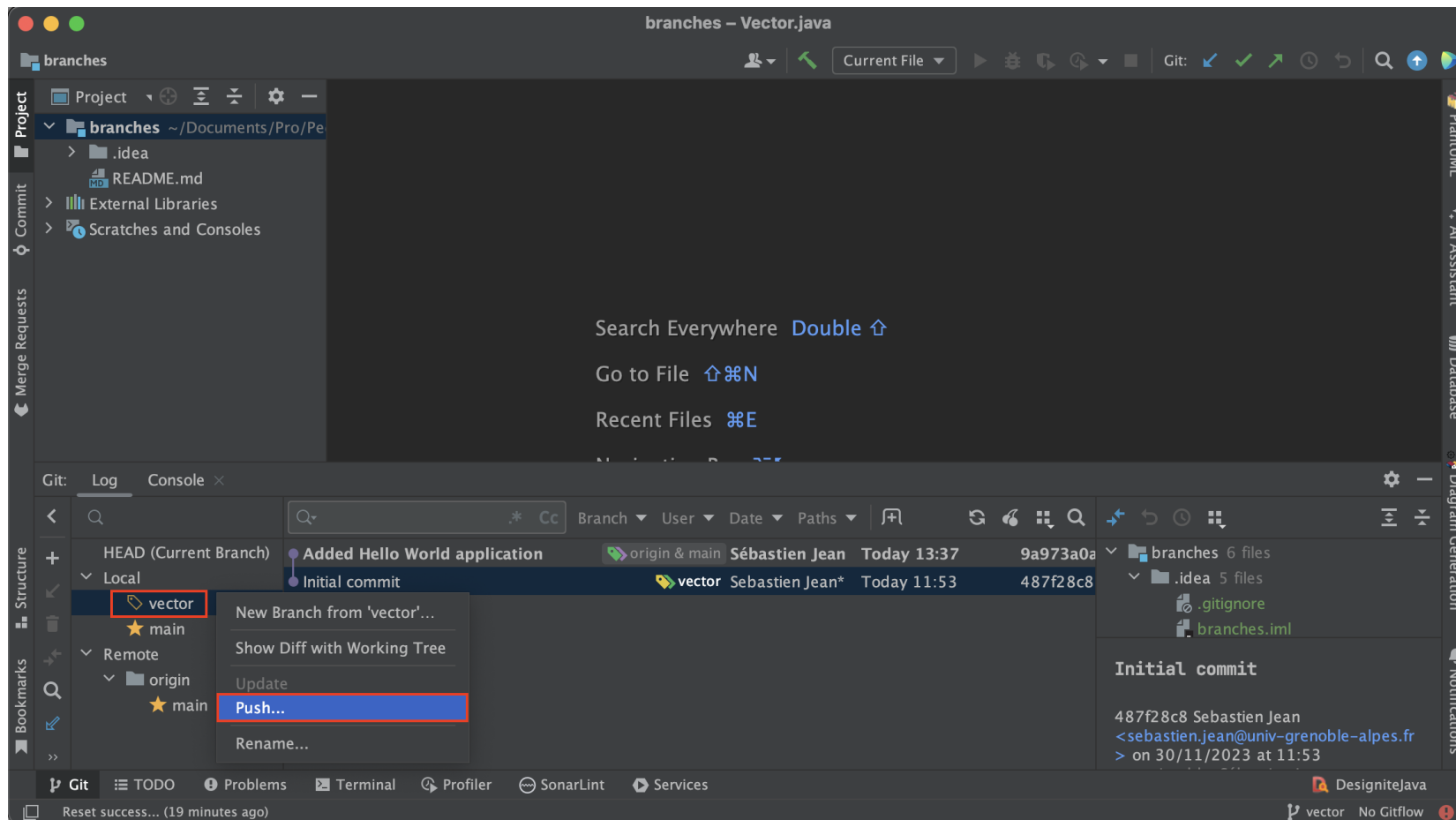
- La branche est créée **localement uniquement**

The screenshot shows the GitLab web interface for a repository named 'branches' (Project ID: 5141). The interface includes a sidebar with navigation options like 'Project', 'Pinned', 'Manage', 'Code', and 'Settings'. The main content area displays the repository's commit history and file structure. A red box highlights the 'main' branch in the branch selector. Below the branch selector, there are buttons for 'History', 'Find file', 'Edit', 'Download', and 'Clone'. A table lists the files in the repository, including '.idea', 'src', and 'README.md', along with their last commit and update times. The 'README.md' file is expanded, showing the title 'Git-Branches' and the description 'Dummy repository for git branching training using IntelliJ.'

Name	Last commit	Last update
.idea	Initial commit	24 minutes ago
src	Added Hello World application	24 minutes ago
README.md	Initial commit	24 minutes ago

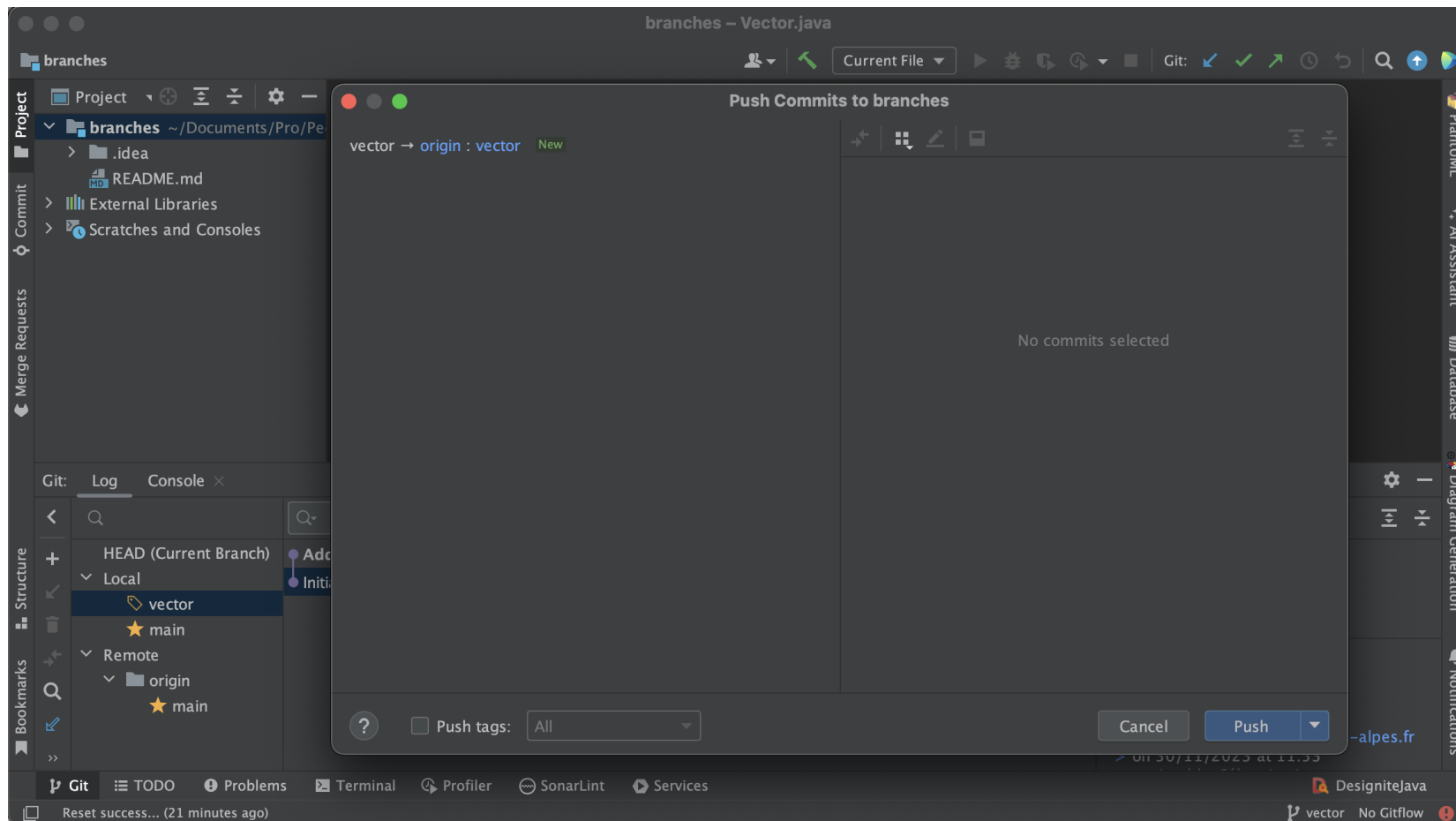
# Publier une branche

- La branche doit être **publiée explicitement** sur le serveur



# Publier une branche

- (La branche doit être **publiée explicitement** sur le serveur)





# Publier une branche, suite

- La branche distante est créée

The screenshot shows the GitLab interface for a repository named 'branches'. The repository has 2 commits and 2 branches. A dropdown menu is open, showing the 'vector' branch being selected. The repository description is 'Dummy repository for git branching training using IntelliJ'. The commit history table is as follows:

	Last commit	Last update
	Initial commit	1 hour ago
	Added Hello World application	1 hour ago
📄 README.md	Initial commit	1 hour ago

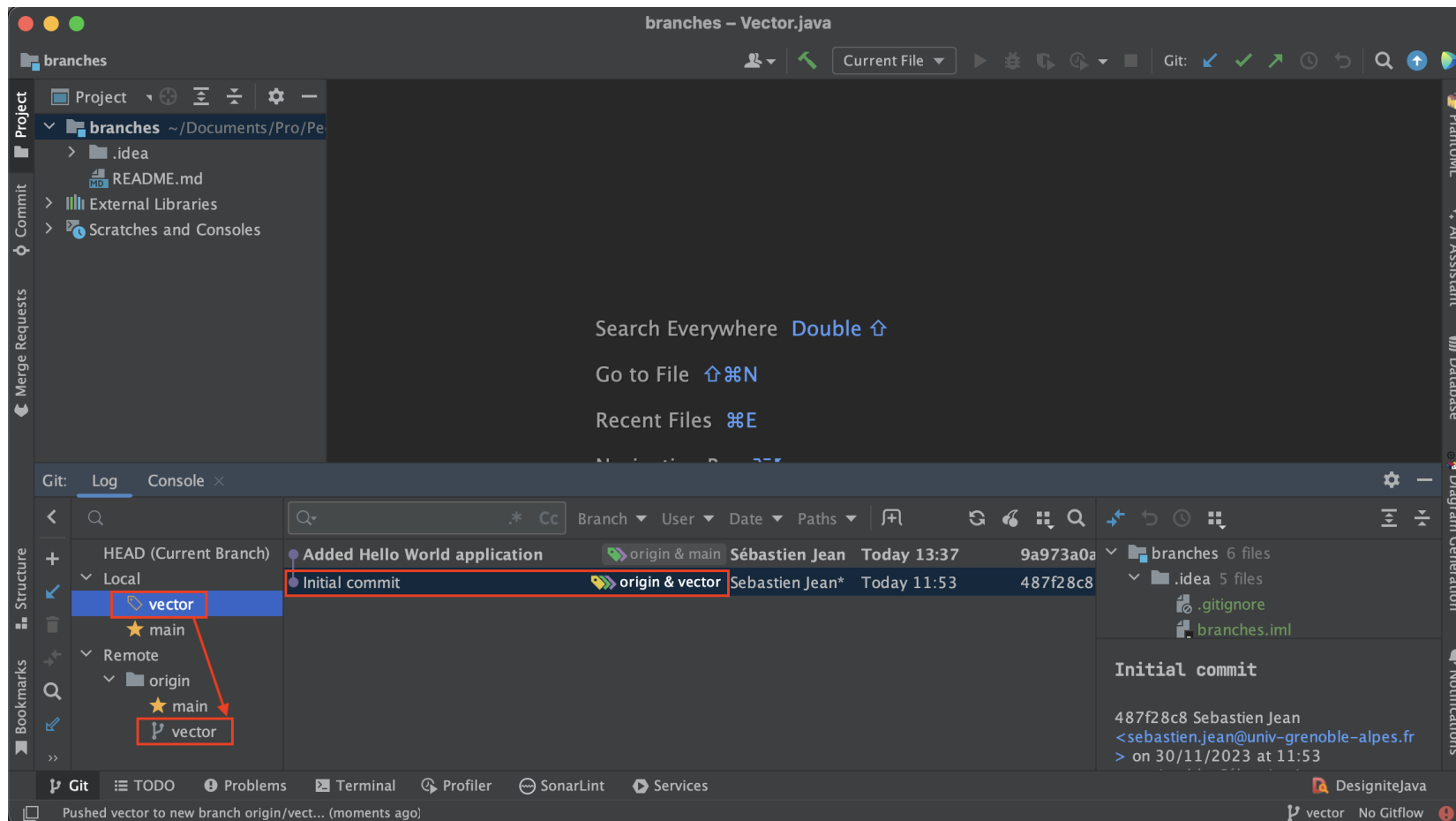
The README.md content is:

## Git-Branches

Dummy repository for git branching training using IntelliJ.

# Publier une branche, fin

- La branche **locale** est **associée** à la branche **distante** (*remote-tracking*)



# Effectuer un commit sur une branche

- Un **commit** sur la branche `vector` fait avancer la branche `vector` **indépendamment** de `main`

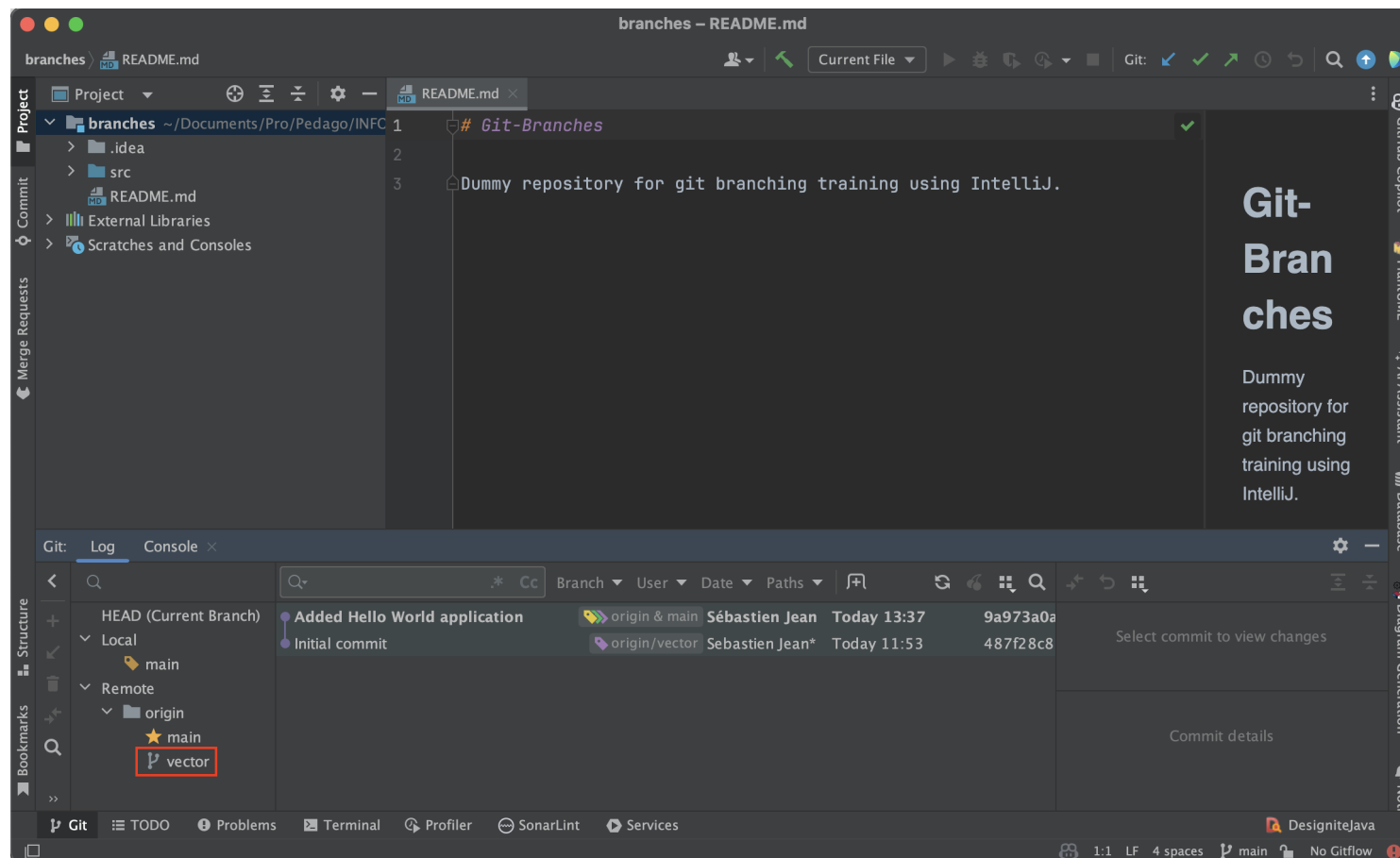
The screenshot shows an IDE window titled "branches - Vector.java". The main editor displays the code for a `Vector` class with two private final double attributes (`cx` and `cy`) and two public getter methods (`getCx()` and `getCy()`). The bottom panel shows the Git log for the `vector` branch, with the following entries:

Commit Hash	Message	Author	Date	SHA-1
97dfb9b9	Added Vector class	Sébastien Jean	Moments ago	97dfb9b9
9a973a0a	Added Hello World application	Sébastien Jean	Today 13:37	9a973a0a
487f28c8	Initial commit	Sebastien Jean*	Today 11:53	487f28c8

The `vector` branch is highlighted in the log, and the commit message "Added Vector class" is also highlighted. The bottom status bar indicates "1 file committed: Added Vector cla... (moments ago)".

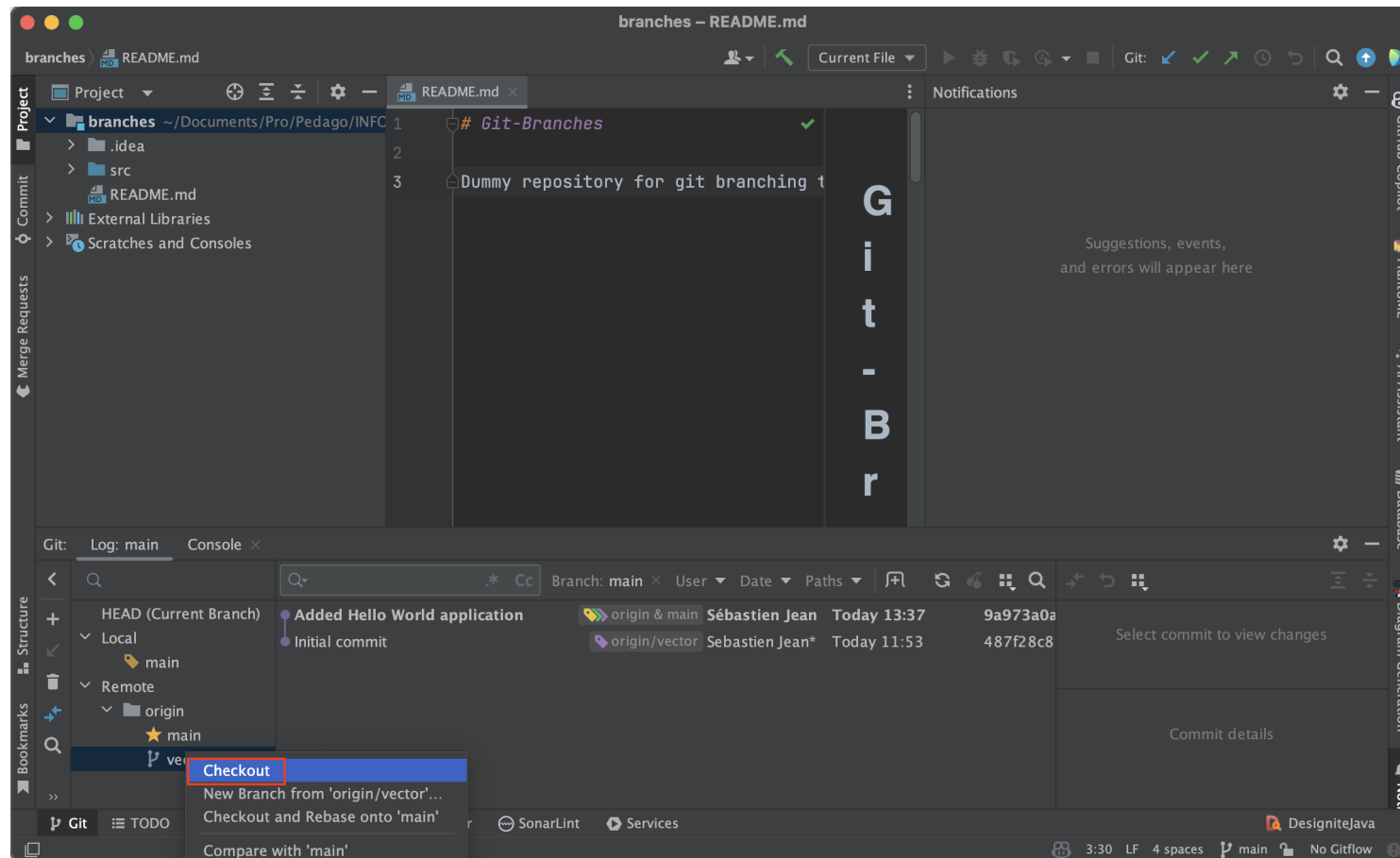
# Cloner et reconstruire les branches

- Le **clonage d'un dépôt** contenant **plusieurs branches** :
  - **rapatrie** toutes les branches distantes (*fetch*)
  - **créé *uniquement*** la branche principale `main`

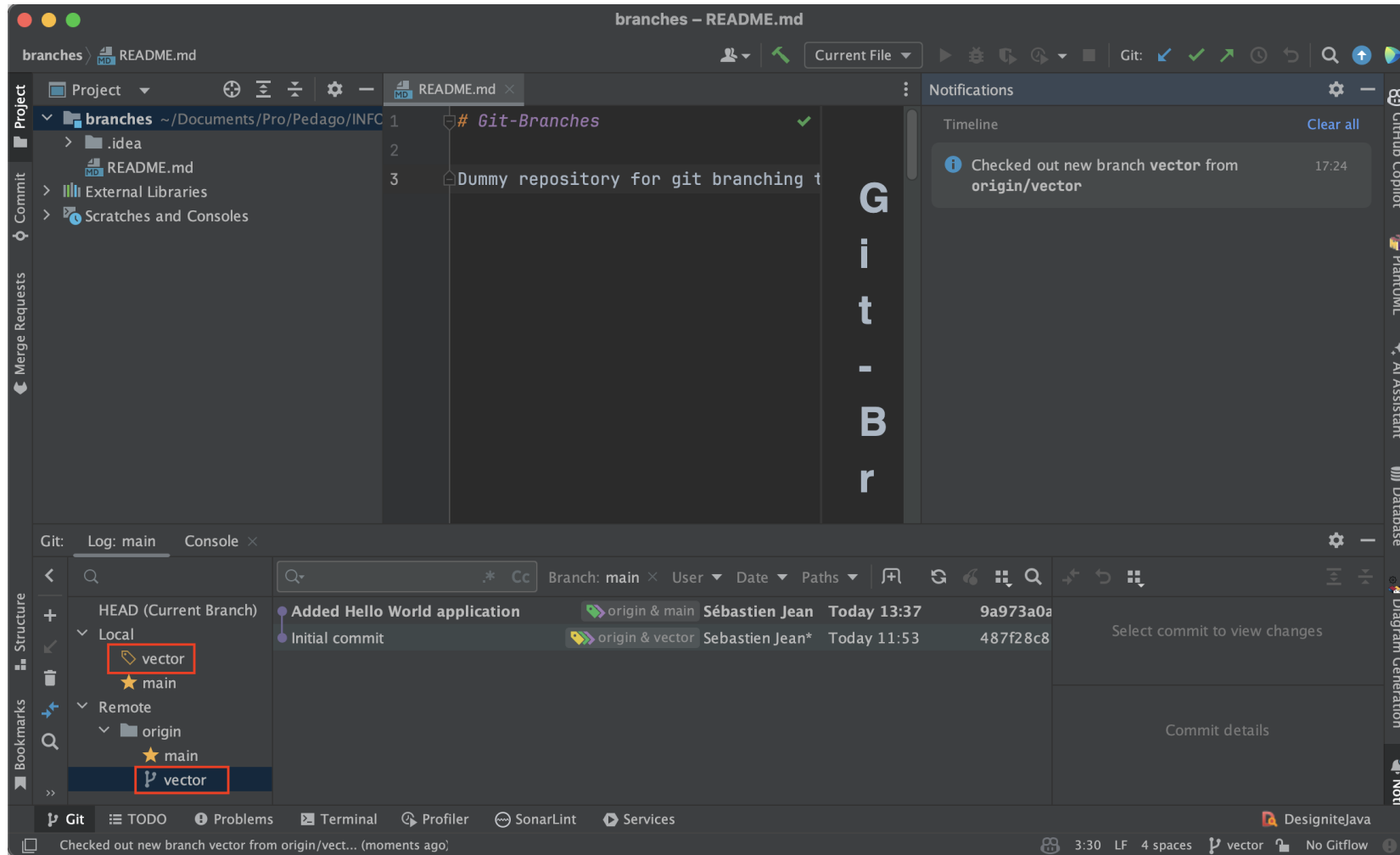


# Cloner et reconstruire les branches, suite

- La création d'une branche locale à partir d'une branche distante est explicite



# Cloner et reconstruire les branches, fin



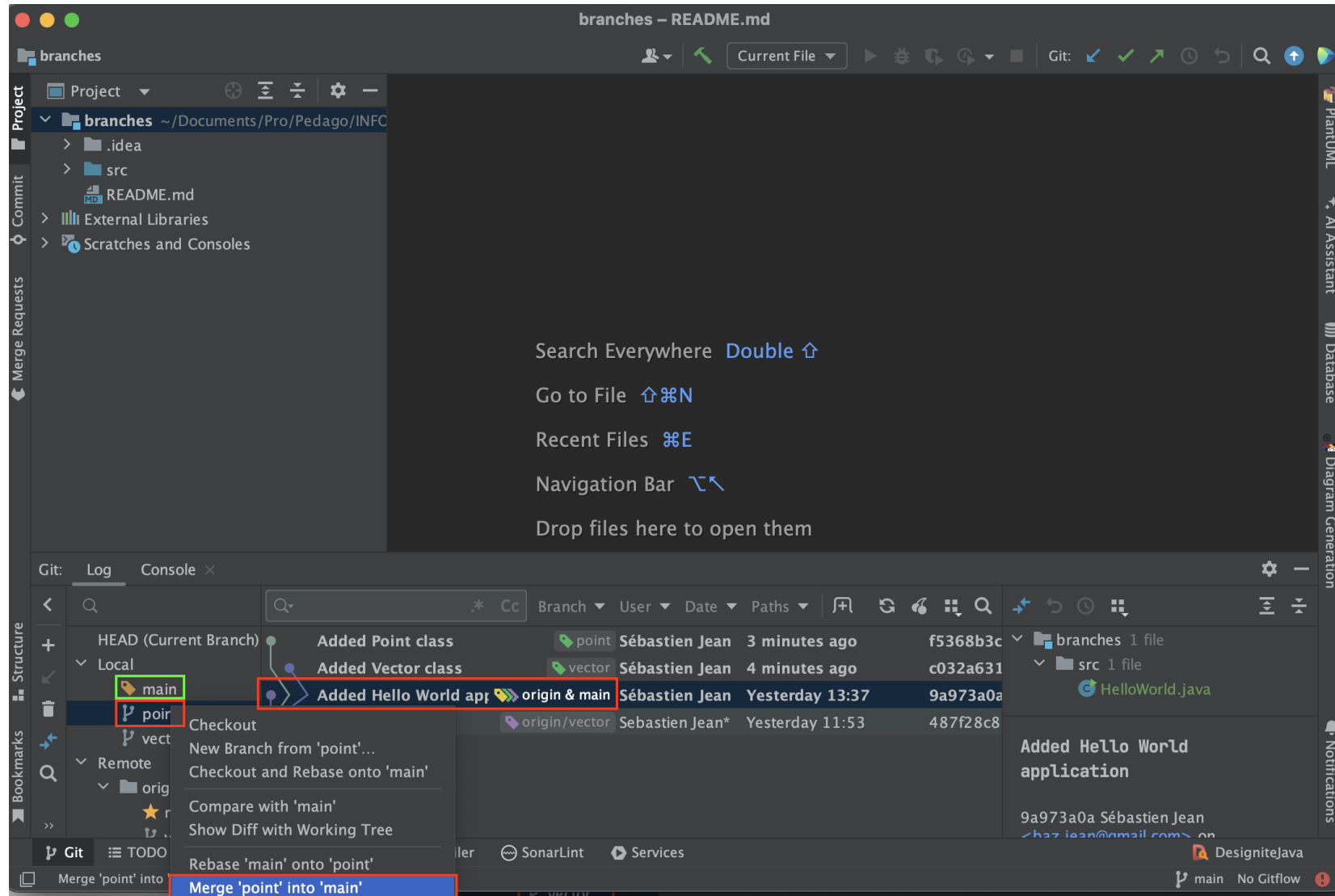
## Création d'une branche, sans publication, et commit

The screenshot shows the IntelliJ IDEA interface with the following components:

- Project View:** Shows the project structure with a local branch named 'point' selected under the 'Local' section.
- Code Editor:** Displays the content of 'Point.java', which contains a Java class definition for 'Point' with two private final double attributes (x and y) and two public methods (getX() and getY()).
- Git Log:** Shows a list of commits. The most recent commit is 'Added Point class' on the 'point' branch, committed by Sébastien Jean. The commit message and details are shown in the bottom right pane.
- Git Status:** Shows '1 file committed: Added Point cla... (moments ago)'.

# Fusionner 2 branches locales (*merge*)

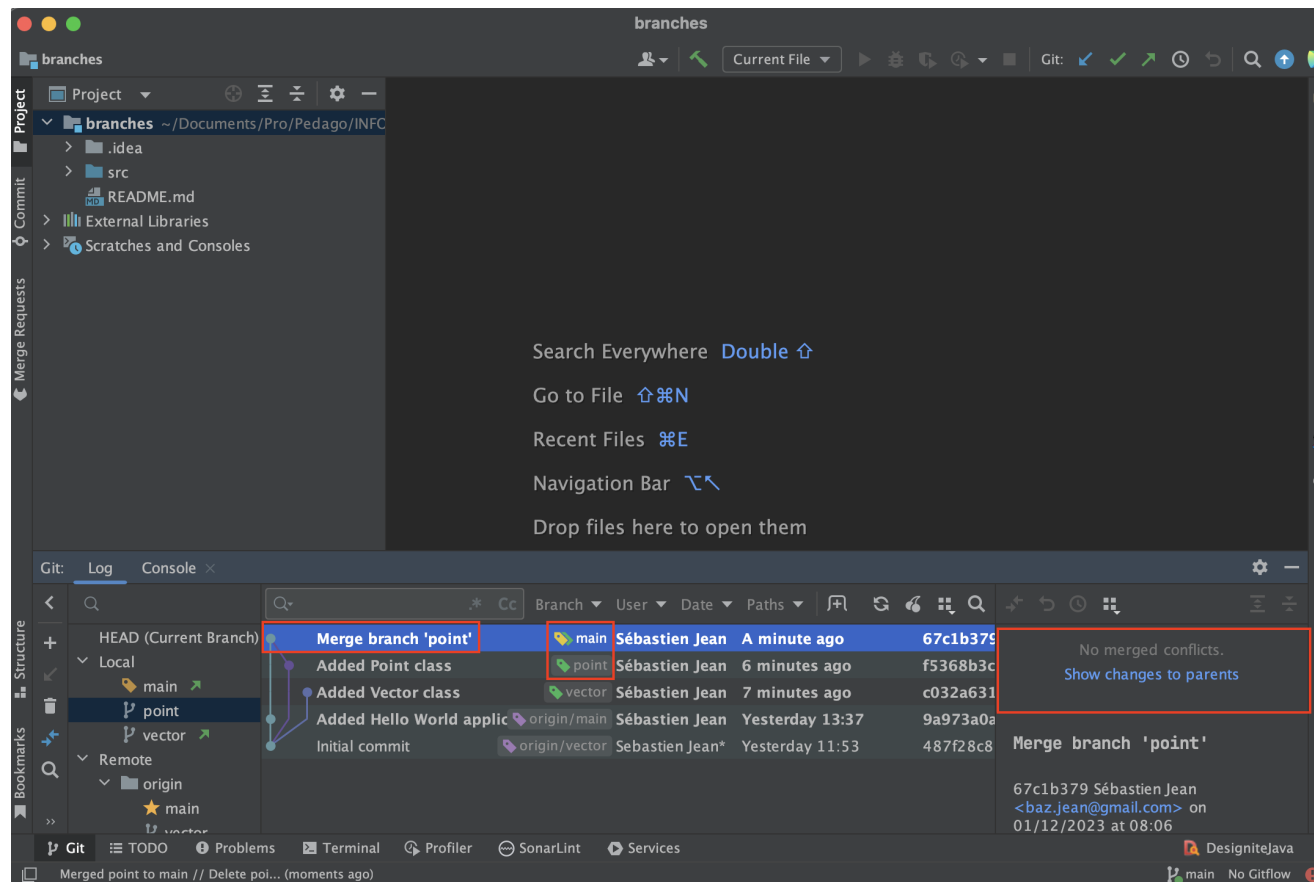
- La fusion s'opère d'une **branche sélectionnée** vers la **branche courante**





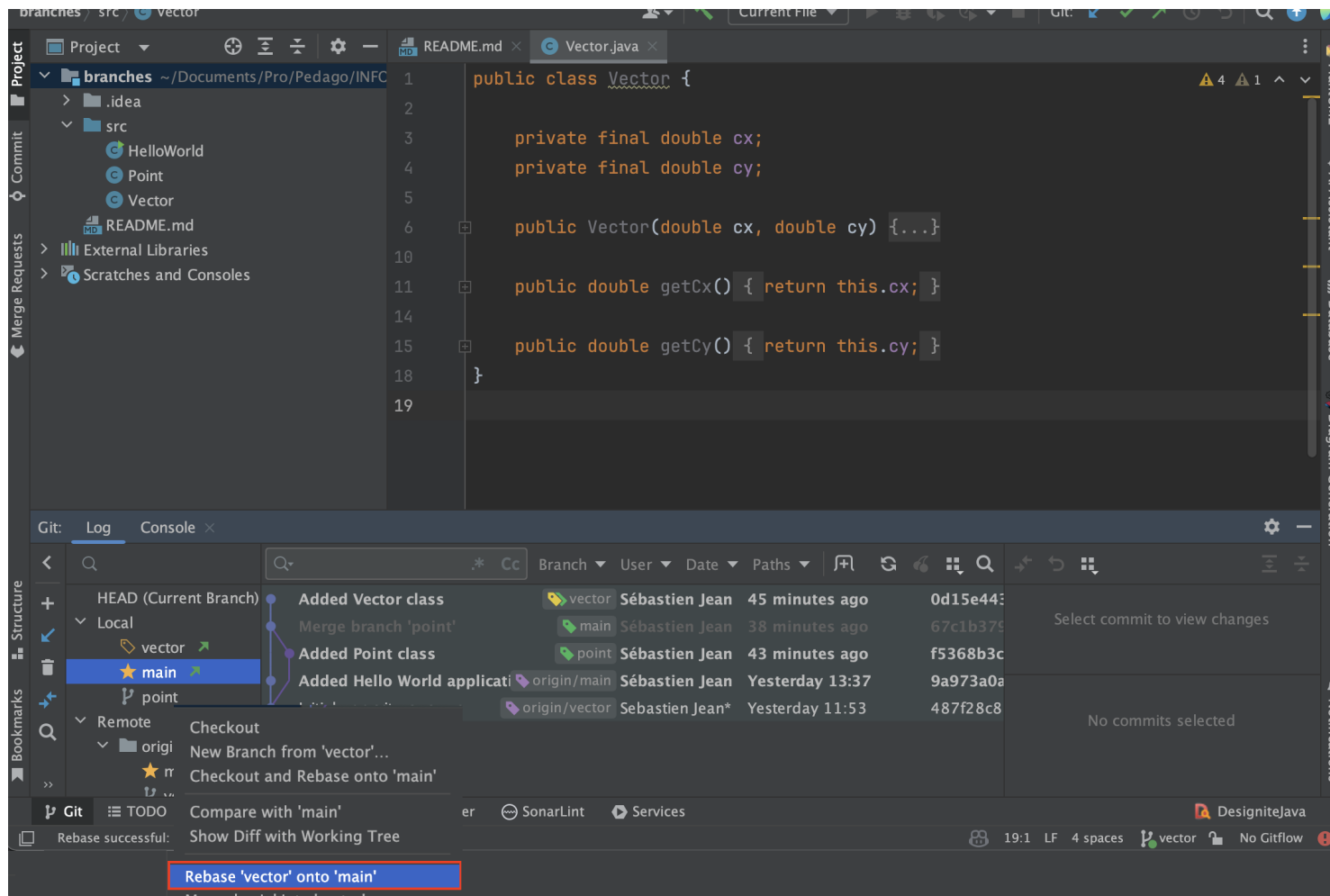
# Fusionner 2 branches locales (*merge*), fin

- La **branche courante intègre** les commits de la branche sélectionnée, **pas l'inverse**
  - La fusion est marquée par un **commit (de fusion) ayant 2 parents**
  - Quand il n'y a **pas de conflit**, le commit n'ajoute pas de modification



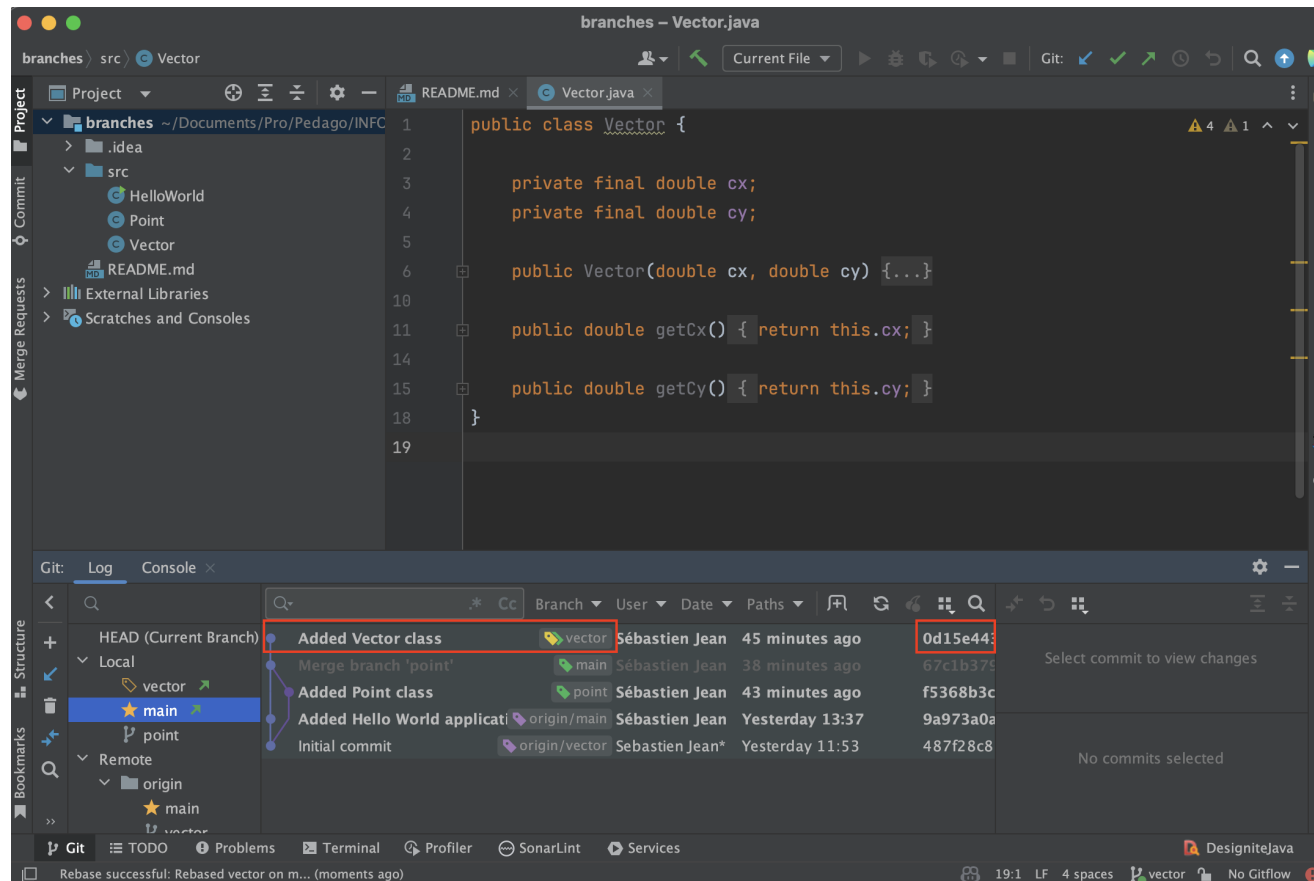
# Rebaser une branche locale (*rebase*)

- Les commits de la **branche courante** sont **recalculés** pour se **replacer au dessus du dernier commit** de la **branche sélectionnée**



# Rebaser une branche locale (*rebase*), fin

- La **branche courante intègre** les commits de la branche sélectionnée, **pas l'inverse**
  - Les commits sont bien recalculés (hash différent)
  - En cas de **conflit**, chaque **commit recalculé** inclura également les **modifications de résolution manuelle**



# Fusionner 2 branches locales (*merge*) sans divergence

The screenshot shows the IntelliJ IDEA interface with the Git tool window open. The commit log displays the following entries:

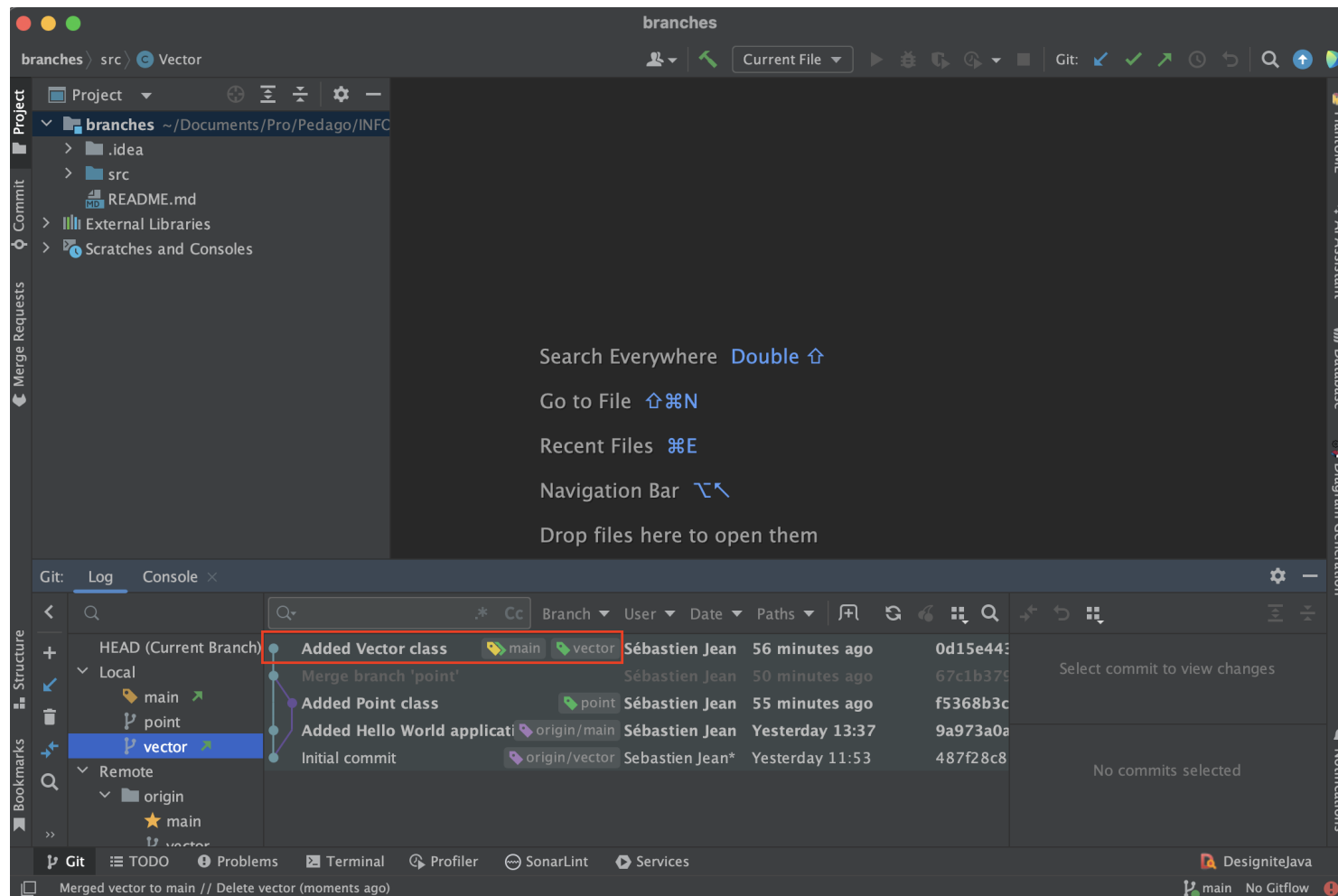
Commit Hash	Message	Author	Date	SHA-1
0d15e443	Added Vector class	Sébastien Jean	54 minutes ago	0d15e443
67c1b379	Merge branch 'point'	Sébastien Jean	48 minutes ago	67c1b379
f5368b3c	Added Point class	Sébastien Jean	53 minutes ago	f5368b3c
9a973a0a	Added Hello World applicati	Sébastien Jean	Yesterday 13:37	9a973a0a
487f28c8		Sebastien Jean*	Yesterday 11:53	487f28c8

The context menu for the 'main' branch is open, showing the following options:

- Checkout
- New Branch from 'vector'...
- Checkout and Rebase onto 'main'
- Compare with 'main'
- Show Diff with Working Tree
- Rebase 'main' onto 'vector'
- Merge 'vector' into 'main'**

# Fusionner 2 branches locales (*merge*) sans divergence, fin

- Le commit de fusion n'est **pas nécessaire**, seul le **pointeur de branche se déplace**



# Picorer des commits (*cherry-pick*)

- Intégrer individuellement des commits d'autres branches
  - Ici, on a créé une branche `cherry-pick` à partir du **commit initial**
  - On y a **intégré deux commits** provenant des branches `point` et `vector`, recalculés

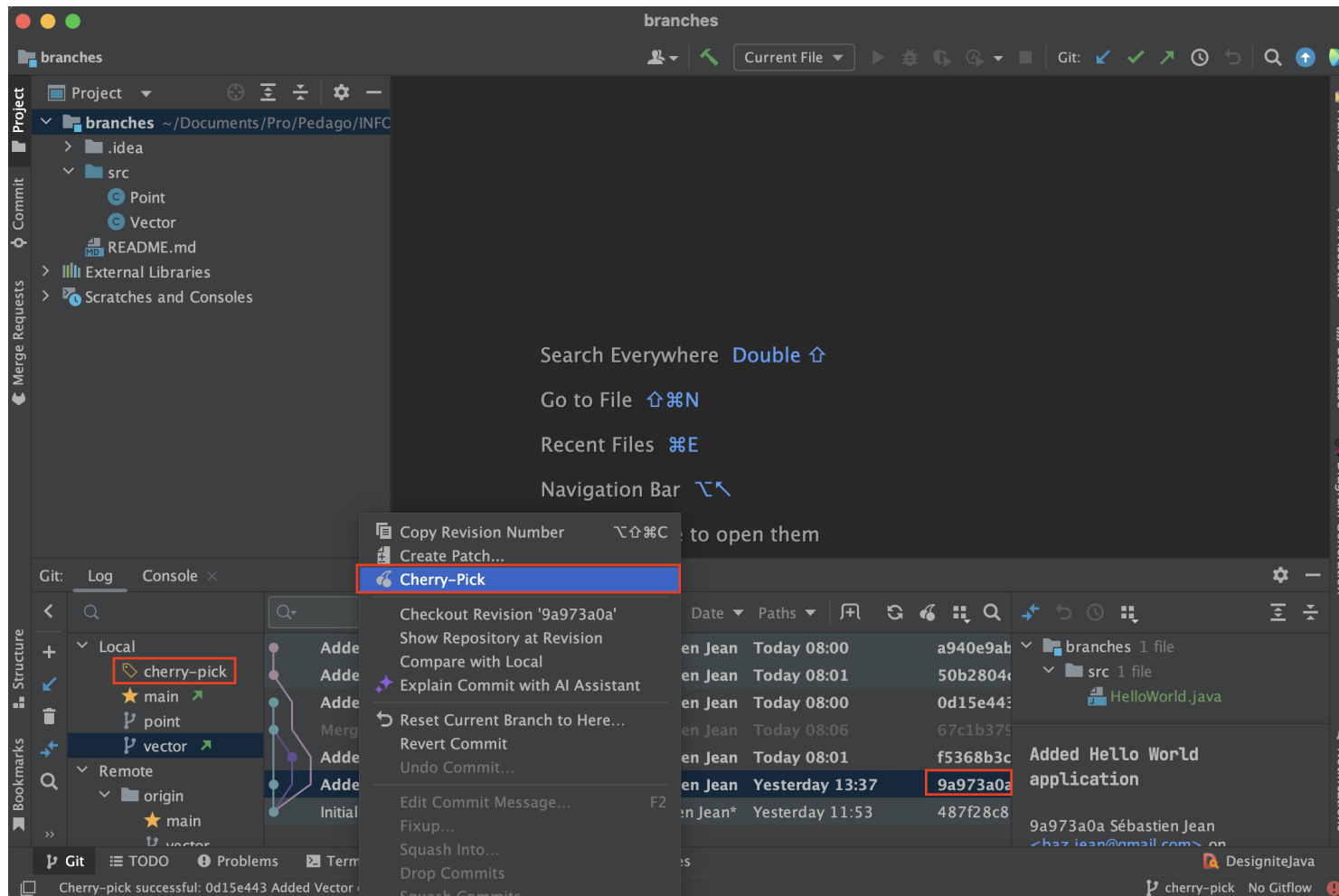
The screenshot displays the Git log for the 'cherry-pick' branch. The log shows the following commits:

Commit Hash	Message	Author	Date
a940e9ab	Added Vector class	Sébastien Jean	Today 08:00
50b2804	Added Point class	Sébastien Jean	Today 08:01
0d15e44	Added Vector class	Sébastien Jean	Today 08:00
67c1b37	Merge branch 'point'	Sébastien Jean	Today 08:06
f5368b3	Added Point class	Sébastien Jean	Today 08:01
9a973a0	Added Hello World applic	Sébastien Jean	Yesterday 13:37
487f28c	Initial commit	Sebastien Jean*	Yesterday 11:53

The IDE interface also shows the project structure with files like `Point`, `Vector`, and `README.md` in the `src` directory. The bottom status bar indicates the cherry-pick operation was successful.

# Picorer des commits (*cherry-pick*), suite

- On intègre un autre commit plus ancien ... ( )



# Picorer des commits (*cherry-pick*), fin

- (On intègre un autre commit plus ancien) ... **recalculé** également

The screenshot shows the IntelliJ IDEA interface during a cherry-pick operation. The 'Log' window is open, displaying a list of commits. The commit 'Added Hello World applic' with hash 9a973a0a is highlighted in blue. The 'Structure' window shows the project files, including 'HelloWorld.java' in the 'src' directory. The status bar at the bottom indicates 'Cherry-pick successful: 9a973a0a Added Hello World applicati... (a minute ago)'.

Commit Message	Author	Date	Hash
Added Hello World applic	Sébastien Jean	Yesterday 13:37	9b3cd024
Added Vector class	Sébastien Jean	Today 08:00	a940e9ab
Added Point class	Sébastien Jean	Today 08:01	50b2804
Added Vector class	Sébastien Jean	Today 08:00	0d15e44
Merge branch 'point'	Sébastien Jean	Today 08:06	67c1b37
Added Point class	Sébastien Jean	Today 08:01	f5368b3
Added Hello World applic	Sébastien Jean	Yesterday 13:37	9a973a0a
Initial commit	Sébastien Jean*	Yesterday 11:53	487f288



# Nouveau point de départ

The screenshot shows an IDE window titled "branches - Main.java". The main editor displays the following code:

```

1 public class Main {
2
3     public static void main(String[] args) {
4         System.out.println(new Point( x: 0, y: 0));
5     }
6 }
7

```

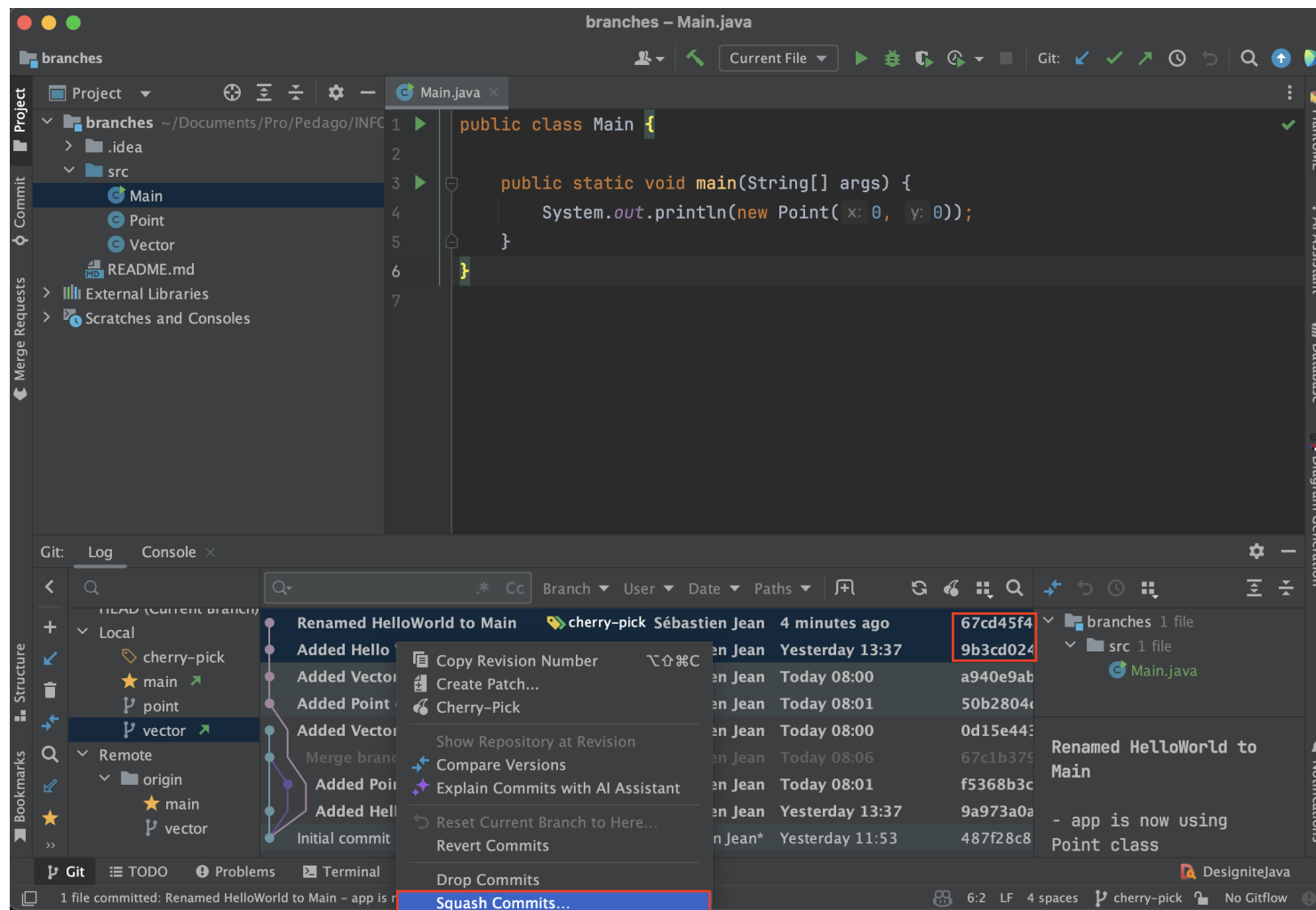
The left sidebar shows the project structure with "Main" selected under the "src" directory. The bottom panel shows the Git Log with the following entries:

Commit Message	User	Date	Hash
Renamed HelloWorld to Main	Sébastien Jean	3 minutes ago	67cd45f4
Added Hello World application	Sébastien Jean	Yesterday 13:37	9b3cd024
Added Vector class	Sébastien Jean	Today 08:00	a940e9ab
Added Point class	Sébastien Jean	Today 08:01	50b2804c
Added Vector class	Sébastien Jean	Today 08:00	0d15e443
Merge branch 'point'	Sébastien Jean	Today 08:06	67c1b379
Added Point class	Sébastien Jean	Today 08:01	f5368b3c
Added Hello World application	Sébastien Jean	Yesterday 13:37	9a973a0a
Initial commit	Sebastien Jean*	Yesterday 11:53	487f28c8

The commit "Renamed HelloWorld to Main" is highlighted with a red box and a "cherry-pick" icon. The bottom status bar shows "1 file committed: Renamed HelloWorld to Main - app is now using Point cla... (2 minutes ago)".

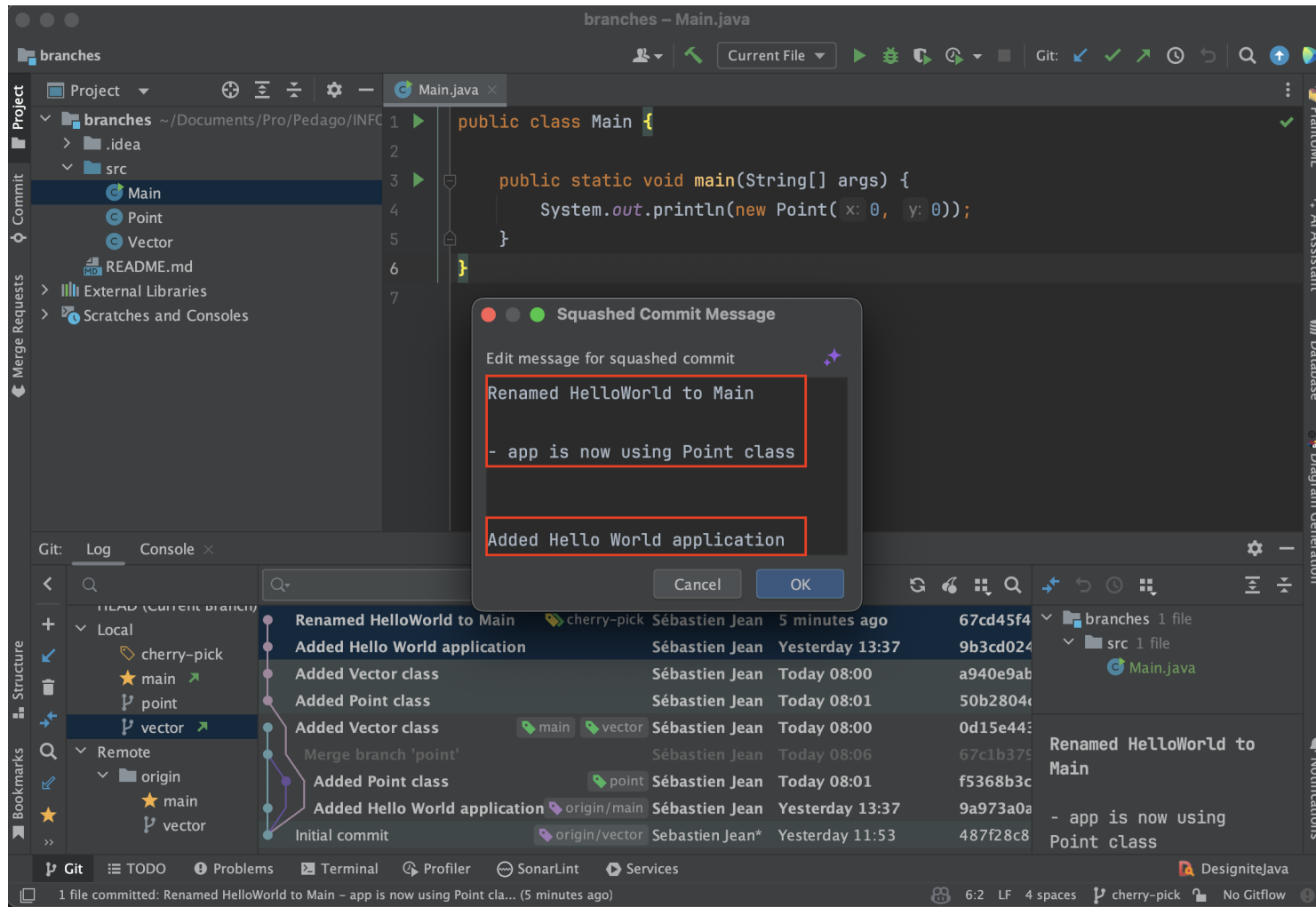
# Regrouper des commits (*squash*)

- **Compacter localement 2 commits** de la **même branche**
- Attention : « interdit » si les commits ont déjà été poussés



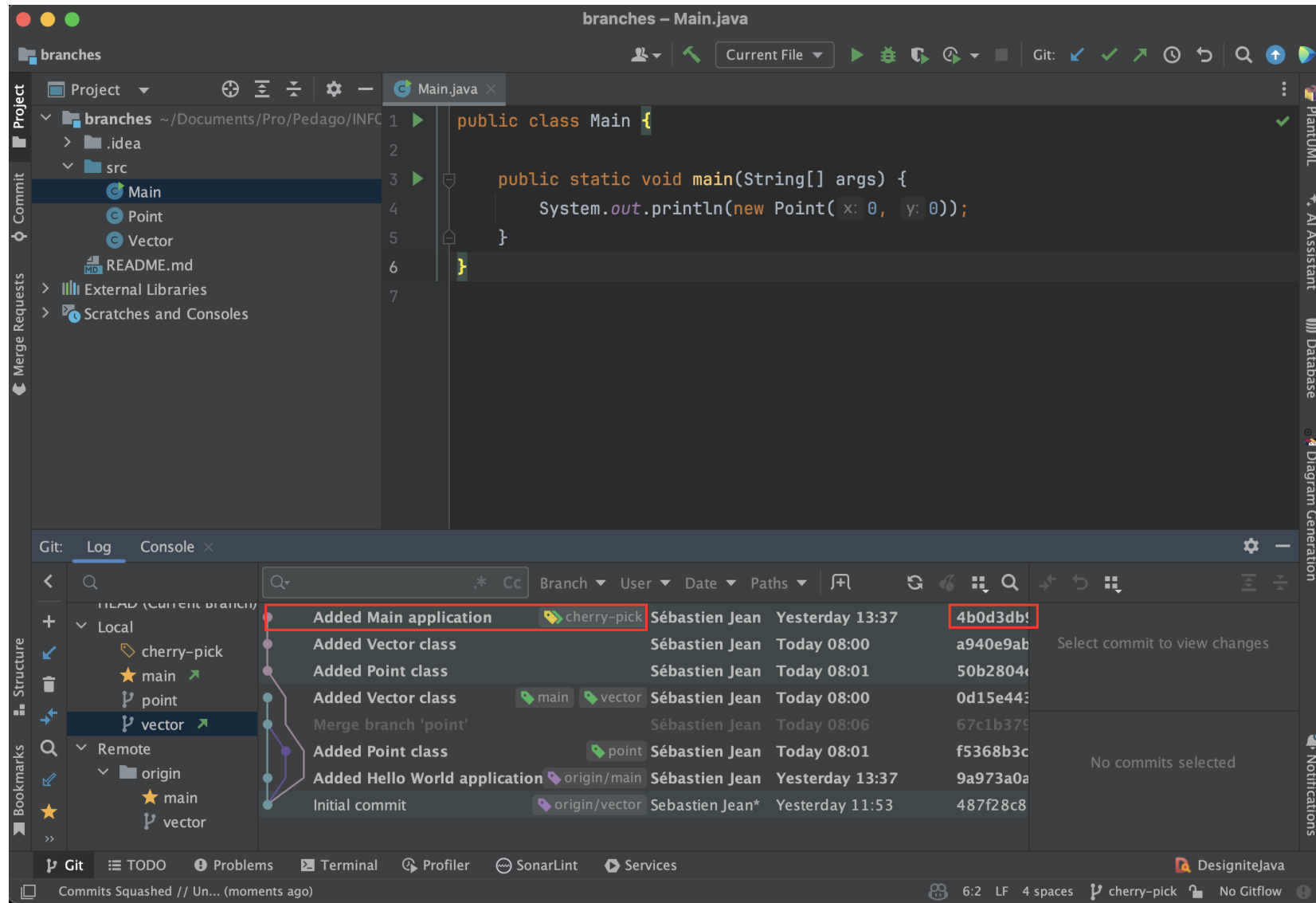
# Regrouper des commits (*squash*), suite

- Le message peut être re-rédigé à partir des 2 messages initiaux



# Regrouper des commits (*squash*), suite

- Le nouveau commit est recalculé

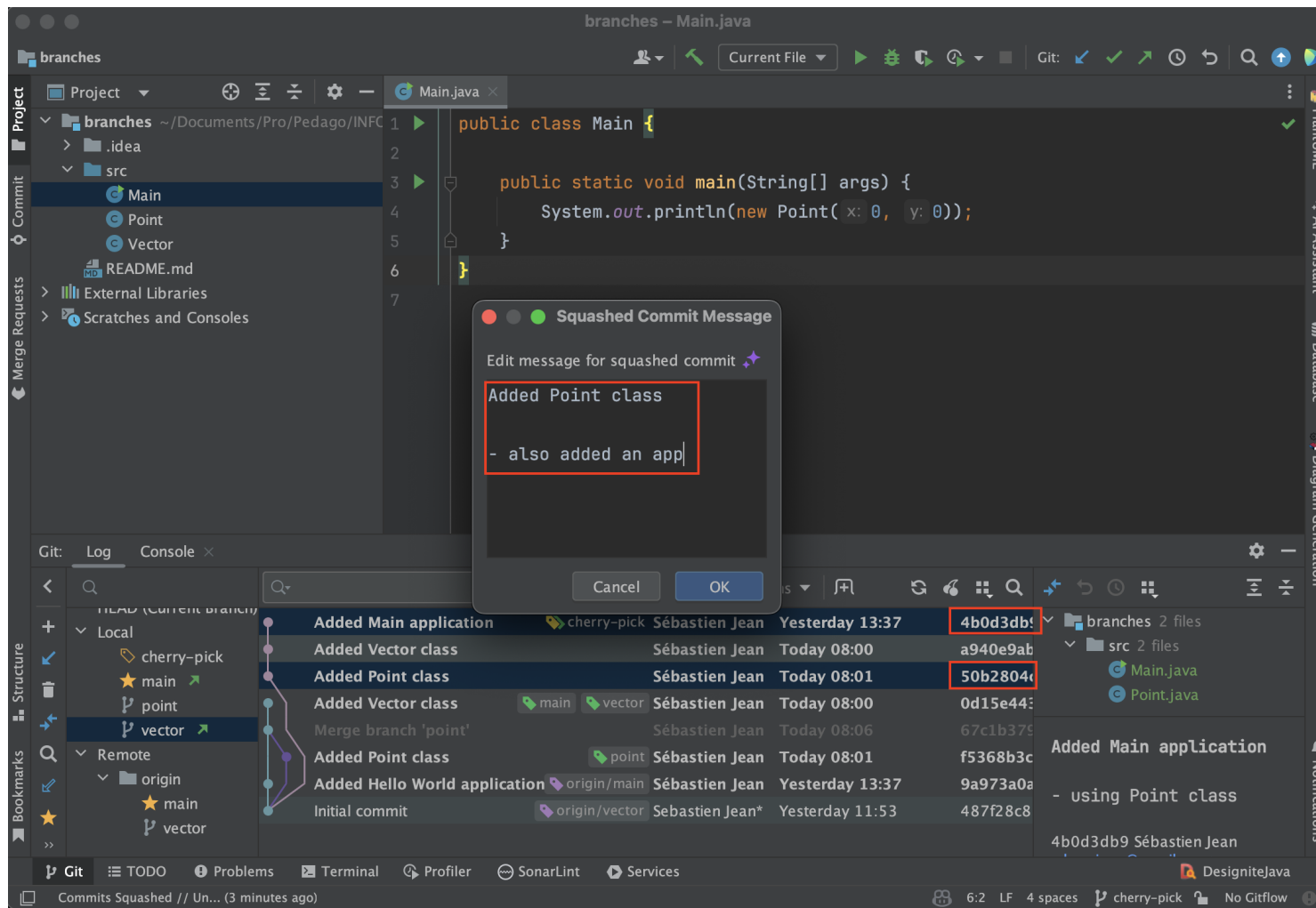


The screenshot shows the IntelliJ IDEA interface with a Git commit log at the bottom. The log displays a series of commits, with the top commit highlighted in red, indicating it is the selected commit. The highlighted commit is titled "Added Main application" and has a commit hash of 4b0d3db1. The log also shows other commits, including "Added Vector class", "Added Point class", and "Merge branch 'point'". The commit log is filtered by the "cherry-pick" action.

Commit Message	Action	User	Date	Hash	Details
Added Main application	cherry-pick	Sébastien Jean	Yesterday 13:37	4b0d3db1	
Added Vector class		Sébastien Jean	Today 08:00	a940e9ab	Select commit to view changes
Added Point class		Sébastien Jean	Today 08:01	50b2804c	
Added Vector class	main, vector	Sébastien Jean	Today 08:00	0d15e443	
Merge branch 'point'		Sébastien Jean	Today 08:06	67c1b379	
Added Point class	point	Sébastien Jean	Today 08:01	f5368b3c	
Added Hello World application	origin/main	Sébastien Jean	Yesterday 13:37	9a973a0a	No commits selected
Initial commit	origin/vector	Sebastien Jean*	Yesterday 11:53	487f28c8	

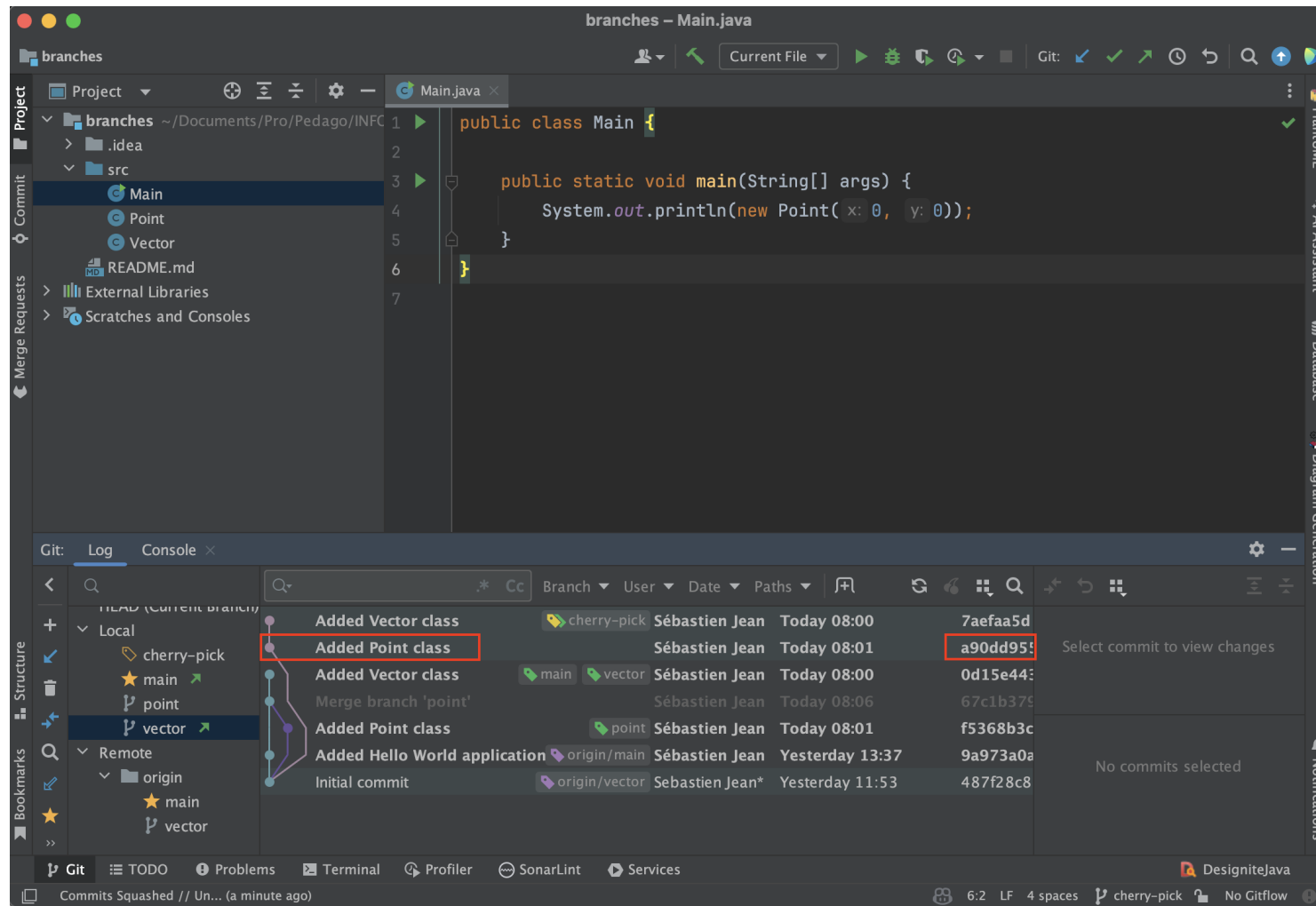
# Regrouper des commits (*squash*), suite

- Les 2 commits ne sont pas forcément successifs



# Regrouper des commits (*squash*), fin

- Le nouveau commit est recalculé
- Attention : il peut y avoir une **résolution de conflit à opérer**



The screenshot displays an IDE interface with a Git commit log at the bottom and a code editor at the top. The commit log shows a sequence of commits, with the 'Added Point class' commit highlighted in red. The code editor shows the content of the 'Main.java' file.

```
public class Main {  
    public static void main(String[] args) {  
        System.out.println(new Point( x: 0, y: 0));  
    }  
}
```

Commit	Author	Date	Hash	Message
Added Vector class	Sébastien Jean	Today 08:00	7aefaa5d	
Added Point class	Sébastien Jean	Today 08:01	a90dd95!	Select commit to view changes
Added Vector class	Sébastien Jean	Today 08:00	0d15e44:	
Merge branch 'point'	Sébastien Jean	Today 08:06	67c1b37:	
Added Point class	Sébastien Jean	Today 08:01	f5368b3c	
Added Hello World application	Sébastien Jean	Yesterday 13:37	9a973a0a	No commits selected
Initial commit	Sebastien Jean*	Yesterday 11:53	487f28c8	

Fin !

