

### Tech Report on Security

In the anti-procrastination app, many security measures must be taken so that you can protect the user and the data.

Some of these measures may include but are not limited to:

- 2FA (two-factor authentication), this could come in the form of a code sent to their email so that extra layer of security is there.
- Strong passwords: the app may require the user to create a particularly strong password, for example it must contain 1 uppercase letter, 1 symbol, and at least 8 characters. This ensures that their password is not easily accessible and can help to prevent unauthorized access to their account.
- Encryption, the app can encrypt login details and users' personal info so that it's protected.
- Auto-logout, if the user is inactive for a certain amount of time while logged into the app, it can automatically log you out in case the user is no longer with their device to prevent unauthorized access.
- Backup Data, the app can backup data at regular intervals so data isn't lost in case the app crashes. In the anti-procrastination app, if the app crashes, you wouldn't want to lose data like, for example, your study schedule for the upcoming week.

It is generally important that the users' privacy is respected and that their personal information is kept private. The app can inform the user of the data that is being collected, and that their data is stored safely using things like access controls to prevent unauthorized access. It is also not uncommon for apps to make sure to get consent from the user to collect their personal information, and the app can do this as well in the form of a message popping up asking them for consent.

Encryption is one of the most important and most common ways to protect user data. In the context of this anti-procrastination app, users enter personal information like their email address, password, etc. This is information that needs to be protected to prevent it from being accessed by attackers. There are many ways in which user data can be encrypted, and one of those ways in which it can be done is by, for example, using AES (Advanced Encryption Standard) which is widely used in various applications, and is even used by the U.S. government to protect sensitive data.

In the anti-procrastination app, users put in personal information like their name, email address and password, personal goals and concerns, things that the user would not want people having unauthorized access to, so these are all important measures that are integral to the app so that users can safely access the app without their personal information under risk, unauthorized access is prevented and data isn't lost in case of the app crashing.

Agile estimations:

Scheduler: Live/Current Schedule - 6 hours

The live scheduler displays the schedule that the user currently has at that point in time, it needs to be linked to the database.

Scheduler: Database - 12 hours

The database contains data of the users' schedules, like what the schedule is, what day it's on, what timeslot, and it contains details of the current schedule, and this will be linked to the interface which will display these things.

Scheduler: Interface - 8 hours

The interface will display the users' schedule for the week, as well as their current/live schedule. The user can also interact with the interface to update/adjust schedules as they wish. I want to make sure the interface looks clear so that the user can easily do this, and can easily identify their schedules to work on.

Scheduler: Live/Current Schedule

 #28 · created 2 weeks ago by Smit Navinkumar

 0

updated 2 weeks ago

Scheduler: Database

 #27 · created 2 weeks ago by Smit Navinkumar

 0

updated 2 weeks ago

Scheduler: Interface

 #26 · created 2 weeks ago by Smit Navinkumar

 0

updated 2 weeks ago

```

drago@DESKTOP-QV9SL1T MINGW64 ~
$ cd team23-22

drago@DESKTOP-QV9SL1T MINGW64 ~/team23-22 (main)
$ git branch MVP_Scheduler

drago@DESKTOP-QV9SL1T MINGW64 ~/team23-22 (main)
$ git checkout MVP_Scheduler
Switched to branch 'MVP_Scheduler'

drago@DESKTOP-QV9SL1T MINGW64 ~/team23-22 (MVP_Scheduler)
$ git add ^[[200~/c/Users/drago/team23-22/src/main/webapp/app~
fatal: pathspec '?[200~/c/Users/drago/team23-22/src/main/webapp/app~' did not match any files

drago@DESKTOP-QV9SL1T MINGW64 ~/team23-22 (MVP_Scheduler)
$ git add C:\Users\drago\team23-22\src\main\webapp\app
fatal: C:\Users\dragoteam23-22srcmainwebappapp: 'C:\Users\dragoteam23-22srcmainwebappapp' is outside repository at 'C:/Users/drago/team23-22'

drago@DESKTOP-QV9SL1T MINGW64 ~/team23-22 (MVP_Scheduler)
$ git commit -a
On branch MVP_Scheduler
nothing to commit, working tree clean

drago@DESKTOP-QV9SL1T MINGW64 ~/team23-22 (MVP_Scheduler)
$ git push
fatal: The current branch MVP_Scheduler has no upstream branch.
To push the current branch and set the remote as upstream, use

    git push --set-upstream origin MVP_Scheduler

To have this happen automatically for branches without a tracking
upstream, see 'push.autoSetupRemote' in 'git help config'.

drago@DESKTOP-QV9SL1T MINGW64 ~/team23-22 (MVP_Scheduler)
$ git push origin HEAD:MVP_Scheduler
warning: redirecting to https://git.cs.bham.ac.uk/team-projects-2022-23/team23-22.git/
Enumerating objects: 646, done.
Counting objects: 100% (646/646), done.
Delta compression using up to 16 threads
Compressing objects: 100% (526/526), done.
Writing objects: 100% (646/646), 978.92 KiB | 1.82 MiB/s, done.
Total 646 (delta 88), reused 636 (delta 82), pack-reused 0
remote: Resolving deltas: 100% (88/88), done.
remote:
remote: To create a merge request for MVP_Scheduler, visit:
remote:   https://git.cs.bham.ac.uk/team-projects-2022-23/team23-22/-/merge_requests/new?merge_request%5Bsource_branch%5D=MVP_Scheduler
remote:
To https://git.cs.bham.ac.uk/team-projects-2022-23/team23-22
 * [new branch]      HEAD -> MVP_Scheduler

drago@DESKTOP-QV9SL1T MINGW64 ~/team23-22 (MVP_Scheduler)
$ |

```

Overview	Active	State	All	Filter by branch name	Updated date	Delete merged branches	New branch
MVP_History					19 1		
f4b4c97a · I created a folder for Hisotry feature part of the app to store all · 36 minutes ago							
MVP_AlarmTimer					0 1		
327a1fdf · branch test · 51 minutes ago							
MVP_Antiprocrastination					0 1		
1ed974f8 · created entirely new branch to try and fix ISSUE · 10 hours ago							
MVP_AntiProcrastination					2 4		
f4ea0640 · pls let this work · 10 hours ago							
MVP_ToDo_List					2 4		
72ab4910 · Update the entity named TodolistItem · 1 day ago							
MVP_Scheduler					0 0		
09342320 · Merge branch 'zxl183-main-patch-58703' into 'main' · 3 days ago							
main	default	protected					
09342320 · Merge branch 'zxl183-main-patch-58703' into 'main' · 3 days ago							

✓ You pushed to MVP\_Scheduler 9 minutes ago

Create merge request

Merge branch 'zxl183-main-patch-58703' into 'main' ...

09342320

Zijun Li authored 3 days ago

MVP\_Scheduler team23-22 / +

History Find file Web IDE

Clone

Name	Last commit	Last update
.devcontainer	Create template	3 weeks ago
.husky	Create template	3 weeks ago
.jhipster	Add blog.jdl files	1 week ago
.mvn	Create template	3 weeks ago
src	Update src/main/docker/app.yml	3 days ago
webpack	Create template	3 weeks ago
.browserslistrc	Create template	3 weeks ago
.editorconfig	Create template	3 weeks ago
.eslintignore	Create template	3 weeks ago
.eslintrc.json	Create template	3 weeks ago
.gitattributes	Create template	3 weeks ago
.gitignore	Create template	3 weeks ago
.gitlab-ci.yml	Update .gitlab-ci.yml file	2 weeks ago