# EGH400 – Project Planning Notes

## Programming language selection

|  |  |  |
| --- | --- | --- |
| HTML, JavaScript with either react, Angular, Vue framework | Java | Kotlin |
| Advantages   * Cross platform (very good for web apps) * Lot of online resources – however there are so many different frameworks and methods to do the same thing finding solutions can be time consuming (in my experience). * Likely will be the least time consuming for gui’s redesign on different sized devices   Disadvantages   * Compatibility of external classes was a real issue in my experience – caused a lot of back tracking. * Error handing is not great – will continue process even with errors (can cause strange behaviour when errors aren’t noticed) * In my experiences with all the files and the ability to do thing in multiple locations can cause code to become needlessly complex / messy * More potential security risks (eg cross site scripting) | Advantages   * Very familiar with language * Already have a lot of the code needed from other projects -could be used to cut down development time (massively). * One of the most popular languages – lots of online resources * Used to be android official language (more than a decade) * Has some nicely flesh out ide’s   Disadvantages   * Not cross platform (no ios) – however will be (jdk is coming to ios but isn’t ready with no data of completion) | Advantages   * New language however is heavily based on java * Cross platform (more directed at mobile apps) * Being heavily push by android (new official language) – likely good language to learn. * Has some nicely flesh out ide’s   Disadvantages   * New language – not a lot of online resources * Never used before * No clue how errors are handled – (but if its based on java its probability fine) |

Kotlin was selected. Use flutter plugin to create ui

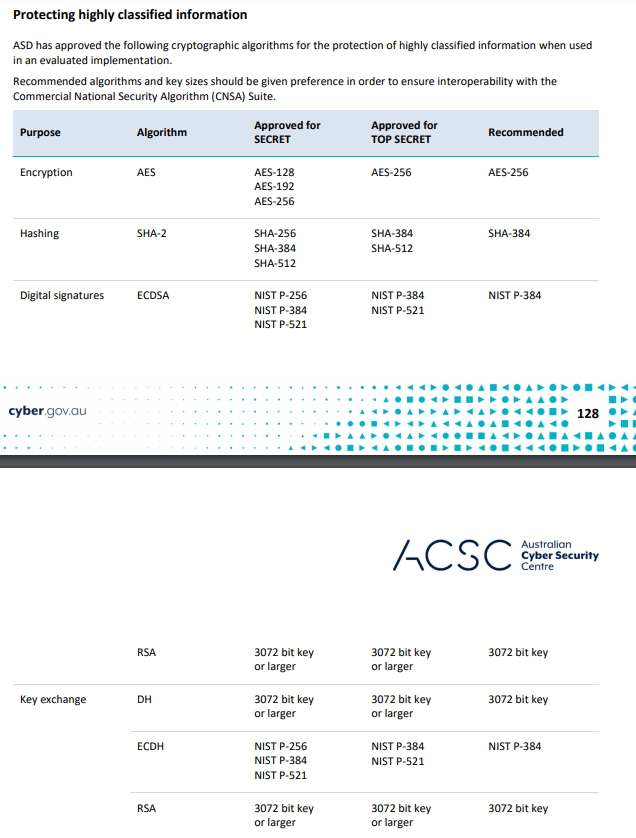
* Can’t make a good case for java -cross platform is a necessity – options to do this are unacceptable.
* I’ve had too many issues with JavaScript frameworks in the past – errors can cause a lot of backtracking making it hard to estimate job completion time.

Note latest versions of flutter are suffering from animation gank (developer have set to low priority issue – won’t be fixed for some time) however app is quite small in scale likely wont need to load huge amounts of animations at start up + login time taken will likely offset the issue anyways

Also note that kotlin does not have socket api for native (ios)

<https://dev.to/jhavidit/integrating-sockets-in-kotlin-4dh5>

## Encryption



<https://www.cyber.gov.au/sites/default/files/2020-01/Australian%20Government%20Information%20Security%20Manual%20%28January%202020%29.pdf>

* Couldn’t find 2021 version however is likely the same.
  + //todo - verify before implementing

## Timeline

|  |  |  |
| --- | --- | --- |
| WK | Tasks | Notes |
| 4 | * Get simple program to run on android and ios * Set up github | * Need to insure kotlin is cross platform main reason for picking over java. * new language – learn basics. – really the main focus for me this week * Never actually put a program onto a device other than desktop – good chance to see if there are any unforeseen issues (eg api incompatibility) |
| 5 | Start app   * Set up client / server sockets * Set up encryption protocol -aes or res 256 | * Jiro uses aes 256 * Refer above for aus standards |
| 6 | * Set up/design database on server * Set up encryption protocol -aes or res 256 – continued | * Likely mysql will do fine   -data on database can be encrypted with sql  -also could simply encrypt data before storing if sql encryption is annoying |
| 7 | * Set up login page   -mainly focus on logic   * Decide on client privileges / roles   -update database |  |
| 8 | * Design home gun / basic layout of apps * Decide on applications of app |  |
| 9 | * Extra week to catch up * No tasks |  |
| 10 |  |  |

## Features desired.

Capability

-organise, sort track team tasks and times from wherever you are

Goal

-better time management, record keeping

Currently used.

DRN outlook calendars, excel, teams and defence diary (no clue what this is – probability pen and paper or word document type thing)

Not a lot of detail is given on actual specifications both on printout and meeting however seems like they only need something simple to organise teams. This could be accomplished in 3 applications.

* Direct messaging
  + between groups and individuals
  + pinging individuals will be needed.
  + Record will need to be stored.
* Visual of tasks/timelines
  + Maybe something like an interactive gnatt chart or list-making (eg Trello)
  + Will need to display states of tasks
    - Completion
    - Users assigned.
    - Time to complete /
    - Ping of incomplete task at certain dates
  + Users will need to be able to present tasks as complete to group (record will need to be kept)
  + Perhaps have an option for personalised view to prevent screen cluttering.
    - Or Individual checklists for users
* Team reports for leaders
  + Report of incomplete jobs
  + Ability to access records.

## Extra tasks

After the above functionality is added extra features will be added

* Desktop interface
* More features / applications
  + Personal notes
* Rotation of screen logic

### Week4