

0.a Come up with a team name for your group.

Code Overflow

0.b Please list the names and PIDs of the team members who are present today (or knowingly absent)

Alan Dial (aland)

Will Logan (willslogan)

Jackson Medina (jacksonm0503)

Joey Kozohar (joeykozohar)

Marcelo Zuleta Sarmiento (zuletamarcelo7)

0.c Provide your preliminary project idea (or set of ideas). This is not a commitment to a project.

Using the approved idea for your group's course project, complete the following activities related to requirements analysis.

- TODO app (keeps track of tasks and prioritizes items)

1. Provide an example of five hypothetical non-functional requirements for this system. Be sure to include the specific type of requirement discussed in class, with each requirement coming from a unique category.

1. Usability: A section in the app that explains how to use our application
2. Reliability: App should be up during peak student hours of the day i.e. 8 am - 5 pm
3. Performance: The app should load fast and not take over 10 seconds of load time.
4. Must store users TODO tasks on their computer (should not be accessible from the Web for security)
5. Implementation: Must be coded in Java

2. Provide an example of five hypothetical functional requirements for this system.

1. Let users check off tasks that have been completed.
2. Allow users to rank how important each task is.
3. Tasks should be encrypted with AES (Advanced Encryption standard), the task list can be decrypted through the user's Password
4. Put reminder for task (set time and date)
5. User should be able to add tasks to the application

3. Think of a specific task required to complete each of the functional requirements and non-functional requirements mentioned above (10 total). Estimate the amount of effort needed to complete this task using function points (i.e., using the values [here](#)). Briefly explain your answer.

8 Points is 1 week of work.

Task	Score
Usability: A section in the app that explains how to use our application	5: Creating a pop window off a press of a button shouldn't require that much effort
Reliability: App should run smoothly and not lag	3: Will need to optimize the app originally but should not be too much changing after first implementation
Implementation: Must be coded in Java	1 : Just need to set up the environment in java
Must store users data locally for security	8: We will need to save tasks to a storage file (json most likely) and there is plenty of support for this kind of task, so it should not take too many points.
The app should load fast and not take over 10 seconds of load time	5: This should require efficient code to minimize runtime.
Let users check off tasks that have been completed	20: The logic behind this might be a little difficult because the initial press of a check mark wouldn't but updating all associated values and how the information is displayed based off the value would be.
Allow users to rank how important each task is	10: Ideally this would be used to sort that tasks, the sorting would likely be handled by the data structure being used to store the information and we wouldn't be writing the code for the sorting.
Tasks should be encrypted with AES (Advanced Encryption standard), the task list can be decrypted through the user's Password	5: Before the task file is read/written to, it will need to be passed through encryption, since this feature isn't very large, it shouldn't take too long.
Put reminder for task (set time and date)	20: This would require implementing a couple of packages and integrating it with our classes could pose some challenge
User should be able to add tasks to the application:	20: This part is condensing all the information that the user presents and storing it in some data structure

4. Write three user stories from the perspective of at least two different actors. Provide the acceptance criteria for these stories.

Story for dev 1:

Implement encryption for the stored tasks file. Criteria, after this story, this task list database file should be encrypted with a password chosen by the user.

Story dev 2:

Design a login page for the app that will take the user's password for database decryption. Criteria, after the story, the user should be presented with a login page when opening the app.

A parent wants to set deadlines for certain tasks or repeated tasks that they do every day. The app will be able to set certain dates for individual tasks with repeatability for those tasks.

5. Provide two examples of risk that could potentially impact this project. Explain how you would mitigate these risks if you were implementing your project as a software system.

One risk that could impact our project is not enough time to meet deadlines. We can reduce this risk by setting goals for when we want to complete certain features and we can delegate tasks between members. Another risk we could face is security. The users will expect that our TODO will keep their tasks confidentially. So we have to find ways to encrypt this data and prevent leaks.

6. Describe which process your team would use for requirements elicitation from clients or customers, and explain why.

We would have a tab on the app that would open a window to a chat bot. The user can then ask questions to the bot and it can assist them as best as possible. If the user is not satisfied with its answers, it can submit a form directly to us where it can describe its problem and suggest additional functionality we should support. Having a chat bot would allow clients to be able to have quick questions they have answered. After the chatbot having a form users can submit would allow us to get direct feedback from our customers in order to improve the functionality of our app.