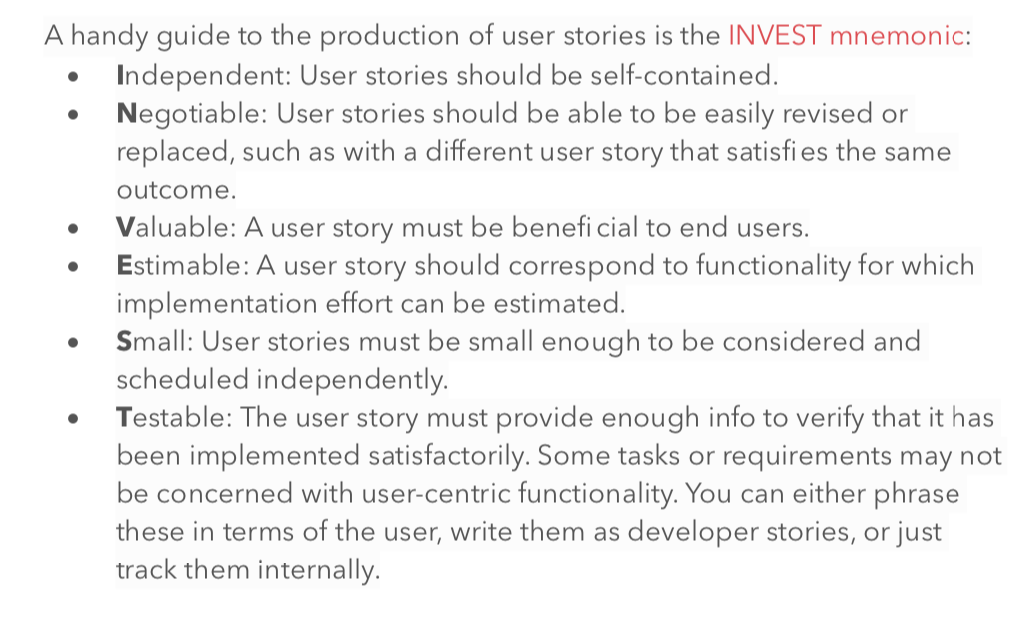
Meeting Agenda for Tuesday (April 28th):

1. Review our work in feature 3-6, raise any questions and try to solve them within the group.
2. Add unsolved questions to the question list below.
3. Combine all use cases and modify our use case diagram.
4. Create a new document (a new google doc) for submission, decide the format.
5. Join Yiwei’s consultation.
6. After the consultation, combine all identified use cases, user stories, summary of achievement and acceptance criteria into the document.
7. Someone record this meeting (meeting agenda and minutes).

Questions for Tuesday’s Consultation:

1. Identify the Use Cases
2. How many use cases should we identify for each requirement of each feature? For example, in **feature 3** we identified 2 use cases, one is creating criteria, the other is selecting the prioritized criteria (we didn’t create different use cases for each criteria, but our user story mentioned specific criteria). Ask for Yiwei’s opinion of this.
3. For **feature 4**, we only identified one main use case, which is finding N number of best candidates, but we have 2 sub use cases. One is contacting the coordinator if the number of candidates is less than the employment requirement. Another is displaying matched candidates if the number is less than the requirement. Should we identify them as 3 individual use cases or just as one main use case and 2 sub use cases? Also, if only one main use case, how many user stories should we create for it?
4. **Feature 5 & 6** (Jasper & Kimberley)
5. Show Yiwei an example of our identified use case and its user stories as well as the summary of achievement and acceptance criteria. Ask for his opinion.
6. Show Yiwei our use case diagram.
7. Do we need a user story for extend or include user case?

**

**Feature 3 (Lucas)**

Use Case 1 - Create Selection Criteria

User story: as an administrator, I want to create selection criteria of range of age, so that only the candidates who are in this range of age can be selected.

Summary of how it is achievable using INVEST:

**Independent**: This user story is independent because it only involves one situation where an administrator creates the range of age criteria for selecting candidates, in this case, it is self-contained.

**Negotiable**: This user story is negotiable because it can be replaced by other cases, for example, the administrator creates the criteria of health record. Both cases aim to select the best candidates for the mission and the same outcome will be fulfilled.

**Valuable**: This user story is valuable because the range of age criteria will select the candidates with most suitable age for the mission, which provides benefit to the mission as well as the administrator (user).

**Estimable**: This user story is estimable because the system can record the age information that the administrator typed and when selecting the candidates, it will check if a candidate’s age is in that range. In this case, it is estimable.

**Small**: This user story is small because the user only needs to create one type of selection criteria and it is independent.

**Testable**: This user story is testable because the information of the N best candidates will be displayed on screen (in feature 5), then the administrator can check whether each of them is in the range of age in the selection criteria.

Acceptance Criteria

1. The administrator will type the youngest age allowed for the mission;
2. The administrator will type the oldest age allowed for the mission;
3. The system should record all the information that the administrator typed and save it as selection criteria;
4. When selecting candidates, the system should check whether a candidate’s age is in the range of age in the criteria so that the candidates with suitable age can be selected.

Use Case 2 - Choose Prioritized Criteria

User story: as an administrator, I want to choose the criminal record criteria as the prioritized criteria, so that the criminal records of candidates will be considered first during selection.

**Independent**: This user story is independent because it only involves one situation where an administrator wants to choose the criminal record as the prioritized criteria, which is self contained.

**Negotiable**: This user story is negotiable because it can be replaced by another case, for example, the administrator wants to select the health record as the prioritized criteria. Both cases aim to provide a prioritized criteria when selecting candidates and the same outcome will be fulfilled.

**Valuable**: This user story is valuable because the administrator thinks that criminal record is the most important selection criteria and it will be considered first during selection. In this case, the administrator’s will be fulfilled thus benefits the user.

**Estimable**: This user story is estimable because if the criminal record is selected as the prioritized criteria, the system will check the candidates’ criminal record first during selection. Then it will check other criteria. In this case, it is estimable.

**Small**: This user story is small because only the criminal record will be selected as the prioritized criteria. The task is small and independent.

**Testable**: This user story is testable because when the system displays the information of N best selected candidates, the administrator can review the criminal record of the selected candidates. Then he can compare it to other candidates’ criminal records to check whether the selected candidates have less (or better) criminal records than other candidates.

Acceptance Criteria

1. The administrator will type the number (N) of the criminal record into the system;
2. The administrator will press Enter to confirm input;
3. The system should save the Nth criteria (criminal record) as the prioritized criteria;
4. When selecting candidates, the system should check the candidates’ criminal record first so that any candidate whose criminal record violates the criteria can not be selected.

**Feature 4 (Lee)**

As an Administrator, I want to type the N number of the best candidates, so that I can have their information to support the mission.

INVEST

**Independent:** This user story is independent because it is not depended on other stories and it can work in any order.

**Negotiable:** This user story is negotiable because it can be replaced by other user stories such as the Administrator wanting to find the detail of candidates for the mission he/she is able to do this method.

**Valuable:** This user story will give the result for the end-user

**Estimable:** This user story is able to show the single function of the task and it is not complex for the end-user to understand.

**Small:** This user story describe a single task function so it is not very complex for end-user to understand

**Testable:** This user story is Testable because the end-user can type any number that they want and the information of candidates will be shown out.

Acceptance Criteria:

1. The administrator is able to type the number (N) of candidates for the mission that he/she wants.
2. The system will filter out the best candidates regarding the mission criteria.
3. The information on the N number of candidates will be shown and then the administrator is able to add them to the mission.

As an Administrator, I want to contact the Mission Coordinator when the number of the best candidates is less than the employment requirement so that the Administrator can discuss with him/her to find a solution.

INVEST

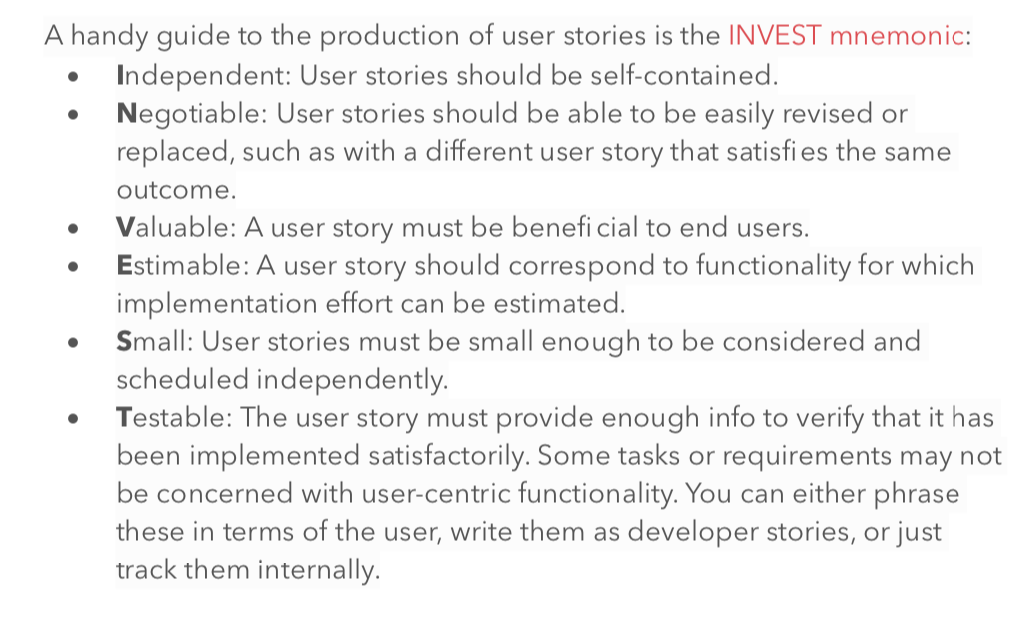
**Independent:** This user story is not independent because it depend on the result from other user.

**Negotiable:** This user story is negotiable because it can be negotiated when team member can provide their solution for the user story.

**Valuable:** This user story is Valuable because the end-user will know what they can do when the number of N candidates can not be sufficient.

**Estimale:** This user story is Estimable because

As an Administrator, I want to display the matched candidates when the number of the best candidates is less than the employment requirement. (Lee: use the first and second user story to write the summary of achievement and acceptance criteria - Lucas)

**

**Feature 5**

Use Case 1 - Selecting candidates

User story: as an administrator, I want to select candidates displayed on the list, so that I can employ my employees.

Summary of how it is achievable using INVEST:

**Independent**: This user story is non dependent because the candidates list depends on the candidates list from finding N best candidates above.

**Negotiable**: This user story is non-negotiable because the selected candidates must be the best N of the list and some of those candidates must be selected to perform the missions.

**Valuable**: This user story is valuable because end users can find out candidates who will be responsible for the mission.

**Estimable**: This user story is estimable because the selected candidates must be in the list.

**Small**: This user story is small because the candidates list has been already generated from finding N best candidates, candidates only need to accept or reject the mission.

**Testable**: This user story is not testable because the system will inform the best N candidates on the list automatically.

Acceptance Criteria

1. The administrator will inform candidates from the selected candidates list and then see which candidates accept or reject the mission.
2. The candidates will only accept or reject the mission.
3. When candidates reject, a replacement of candidates will be generated automatically.
4. When all the candidates accept to go to the mission, the mission coordinator will be informed.

**Feature 6**

Use Case 1 - Creating profile

User story: as a candidate, I want to create a profile , so that I can register as a candidate.

Summary of how it is achievable using INVEST:

**Independent**: This user story is dependent because the candidate can create his or her own profile without relying on other features.

**Negotiable**: This user story is not negotiable because only when a candidate registered can he or her continue to employ for the missions and complete the missions.

**Valuable**: This user story is valuable because the end user can register as a candidate and participate in various missions.

**Estimable**: This user story is estimable because registering as a candidate is a normal entry of the system.

**Small**: This user story is not small because it requires much information of the candidate and also requests a third-party to get the candidate’s criminal and health records.

**Testable**: This user story is untestable because the system will inform the best N candidates on the list automatically.

Acceptance Criteria

1. The candidates available in the module have been interviewed, The candidates will enter the information for creating his or her profile.
2. A third-party is going to get the candidate’s criminal and health records.

**Feature 6**

Use Case 2 - Edit profile

User story: As a candidate, I want to edit my profile, so that my profile is real time data.

Summary of how it is achievable using INVEST:

**Independent**: This user story is non dependent because the profile can be modified after it has been created.

**Negotiable**: This user story is non-negotiable because the profile can not be modified in other user stories.

**Valuable**: This user story is valuable because the end user may join the mission by editing their profiles.

**Estimable**: This user story is estimable because the profile must be the things asked to be provided.

**Small**: This user story is small because the candidates only modified their profile.

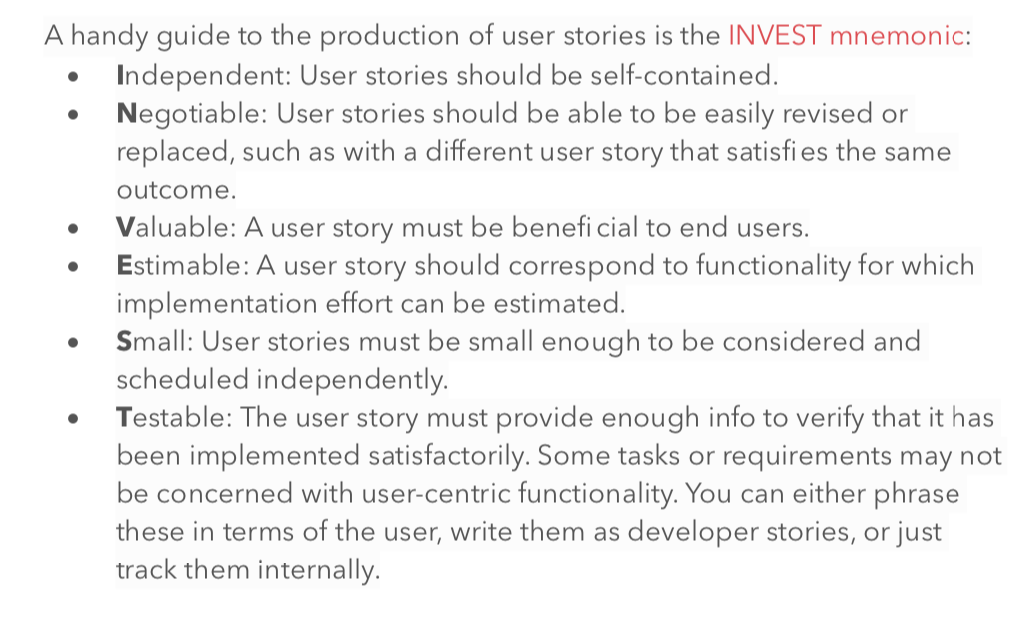
**Testable**: This user story is testable because the result will show after the candidates can modify their profiles.

Acceptance Criteria

1. The candidates can type their information when modifying their profiles.
2. The candidates can type submit to save their data.
3. The system will record the data used for choosing best N candidates.

Follow this structure：As a , I want so that

*Example: as a survey participant I want an indication of progress so that I know how much I have left to complete.*

**

Feature 5

Feature 6