UNIVERSITY OF ALBERTA

Meeting Agenda and Minutes

Agenda Items

1. Introductions
2. Discuss Project / Scope
3. Clarifications / Q&A

Attendees:

Jensen Khemchandani

Peter Qiu

Ronggang Cui

Matthew Sheydwasser

Leen Alzebdeh

Victor Nguyen

Tomas Peschke

tchristie@zerorampup.com

jotoole@zerorampup.com

Introductions

Notes:

- Became acquainted with Jeff and Taylor

Discuss Project / Scope

Ask priority and content of user stories / requirements (We can refine this during Sprint 2 meeting)

Notes:

- **Glossary terms:**
  - "Dev in a box" Administrator – The people on the site that can register and create/view "dev in a box" environments
  - "Dev in a box" User - The people who have access to a "dev in a box" environment.

- **Requirement Priority**
    1. Create administrator account
    2. Login
    3. Payment details (enter on account creation)
    4. Create environment
    5. View environments

- **Administrator register/login**
    - Two methods of registering/logging in supported: – Don't worry about creating our own registration system
        - Github
        - Google OAuth
        - Thought process is that this makes things simpler and focuses on ease of use for the developer

- **Dashboard - manage 'dev in a box instances'**
    - **What should the user see when they log in?**
        - **Login (first time) (automatically login on register):**
            - Add credit card information
        - **Login (general)**
            - If existing dev in a box associated with account, list out existing instances and details on a dashboard
            - If no existing 'dev in a box' associated with the account:
                - Direct them to page to create a new dev in a box and carry out payment

    - **How can dev in a box be managed?**
        - When a box is created – spit out url/passwords and other important info needed to access
            - As an instance is launching – poll the details to get the status of the instance and maybe report that on the front end? – could be a 'nice to have' – low on priority for now
        - Need to be able to view dev in a box details
            - There is a details endpoint in the mock API
            - Important information about creating a "dev in a box" is the 'id' param which can be passed into the details api to get information and data about a specific box
        - Possibility that certain parameters can be changed?
            - Right now things seem pretty immutable
            - Maybe a 'slider' or number representing the amount of developers that can use the "dev in a box". (instances)
                - This is low priority right now – can figure out details late into the project
    - **How are administrators and "dev in a box"'s related**
        - Each dev in a box will have a user associated with them – mock API will include being able to get all dev in a boxes that have a certain user connected to them

- Mock API is fully documented – will have access when we have an NDA signed
    - Is essentially a file that tells us what we can and cannot send to the system and what we can expect to receive.

- **Payment processing**
    - Payment should be done 'behind the scenes' – payment is processed as part of adding a 'dev in a box' – Put in payment information at registration
    - Payment is a subscription (monthly subscription – only one payment plan at the moment with a fixed rate)
        - Using 'Stripe' as the backed solution for the payment processor
    - People can spin up more than one dev in a box environment – payment would differ depending on how many workstations of dev in a box a user has.
        - For now, worry about a single recurring subscription fee per dev in a box – flat fee
            - If you want to set up two "dev' in a box", the subscription would be charged to the administrator twice.
    - Payment information is done via credit card information– see Stripe for details

New action items and responsibilities:
- Create new user stories / requirements for anything discussed in the notes above
    - Include any new functionalities and features in the high level design
- Include 'Stripe' in high level design document as a third party payment processor
    - Research Stripe documentation
- Ensure the fact that a user can have multiple "dev in a box" environments is reflected in any technical/non-technical documentation

Clarifications / Q&A / Planning

Notes:
- Fridays are always the best for meetings
    - Can usually slip in meetings here or there during the work/school day if needed.
- Most simple forms of communication (i.e., simple clarifications and questions) can be done via Slack
- Schedule a new meeting for after the first (or second?) sprint – when we have something new to show the clients.

New action items and responsibilities:
- Ensure that everyone is invited to the Slack group and has access to any additional lines of communication

Additional references

Meeting planning doc:

https://docs.google.com/document/d/1njF8BJ0sVlRqsJjd4T19kchIKv22jY53T7AvbGpDZ0g/edit