March 5, 2018

Eloniah

Talia Santos | Thanh Hoang | Chanelle Hart | Dholon Akter

.

SETUP DOCUMENT

v1.0

Contents

[Agreements 2](#_Toc508005822)

[Processes 2](#_Toc508005823)

[Functional Requirements 2](#_Toc508005824)

[GUI 2](#_Toc508005825)

[Wireframe 3](#_Toc508005826)

[ERD 3](#_Toc508005827)

# Agreements

made with the client

# Processes

Describe all meaningful interactions between the user and the system

Find a systematic approach to do so *(hint: use-cases)*

# Functional Requirements

Group them by application/website

Prioritize them using MoSCoW

MoSCoW method:

Must have

Requirements labelled as Must have are critical to the current delivery timebox for it to be a success. If even one Must have requirement is not included, the project delivery should be considered a failure (note: requirements can be downgraded from Must have, by agreement with all relevant stakeholders; for example, when new requirements are deemed more important). MUST can also be considered an acronym for the Minimum Usable Subset.

Should have

Requirements labelled as Should have are important but not necessary for delivery in the current delivery timebox. While Should have requirements can be as important as Must have, they are often not as time-critical or there may be another way to satisfy the requirement, so that it can be held back until a future delivery timebox.

Could have

Requirements labelled as Could have are desirable but not necessary and could improve user experience or customer satisfaction for little development cost. These will typically be included if time and resources permit.

Won't have (this time)

Requirements labelled as Won't have been agreed by stakeholders as the least-critical, lowest-payback items, or not appropriate at that time. As a result, won’t have requirements are not planned into the schedule for the next delivery timebox. Won't have requirements are either dropped or reconsidered for inclusion in a later timebox. (Note: occasionally the term Would like to have is used; however, that usage is incorrect, as this last priority is clearly stating something is outside the scope of delivery).

Variants

Sometimes W is used to mean Wish (or Would), i.e. still possible but unlikely to be included (and less likely than Could). This is then distinguished from X for Excluded for items which are explicitly not included.

# GUI

Image and description

# Wireframe

Provide a brief description regarding the design

# ERD

Provide a brief justification regarding the design