

3rd meeting with mentor

20.02

Work Organization

- Meetings should be scheduled in the calendar.
- One-to-one meetings should be completed by next Thursday.
- Move to prototyping over the weekend.

Requirement Gathering

- How to ensure there are no issues with requirement collection?
 - Work with Figma prototypes.
 - Clarify overall scope.
 - Create a feature list with priorities and estimated completion times.
 - If estimates are unclear, break tasks into smaller parts (WBS may help).
 - Time estimates might not be accurate.
- Sending individual features to the client is not ideal; better to send in functional parts.
- Should we start with KIP or Bricks School?

Project Management

- Divide the project into stages.
- When will team processes be stabilized?
- Are retrospective sessions in place?
- How to assess our current progress?
- When can we finalize our processes?
- SCRUM is not just about development.
- **Purpose of daily standups:** track progress.
 - What was done?
 - What will be done?
- Set up task trackers (Kanban).

Requirement Definition and Validation

- Is "Define and Validate Project Requirements" part of strategic planning?
 - Probably yes, but it sounds too broad.

- Does "platform" refer to the technical component, not the entire system?

Testing and Metrics

- Unit tests will be integrated into the MVP.
- Usability can be measured by task completion time.
- Evaluating usability through documentation is not the best approach.

System Requirements

- **Availability:**
 - 90% uptime with scheduled maintenance windows communicated in advance.
 - Validate this assumption.
- **Performance:**
 - The platform should support 500 simultaneous competitions.
 - Avoid generic "no degradation" statements; measure performance levels instead.
 - Response times for competition creation and user submission viewing should not exceed 3 seconds.
 - Determine if these are separate requirements.
- **Feedback Mechanisms:**
 - Users should have a way to report usability issues.
 - This seems like a functional requirement.
- **Scalability:**
 - The platform should scale to accommodate increasing numbers of competitions and participants.
 - More about architecture than requirements.
- **Security:**
 - Protection against SQL injection, XSS, and other common threats.
 - Should this be included in KIP or removed?

Additional Tasks

- Tests should align with quality attributes.
- Review quality models and choose an appropriate one.
- At the end of the course:
 - Show MVP this semester.
 - Further improvements will follow.
 - MVP alone may not fully satisfy the client.

- Align client expectations accordingly.