Quality and Risk Management

TakuMahi Evidence Review Four

Review Itinerary

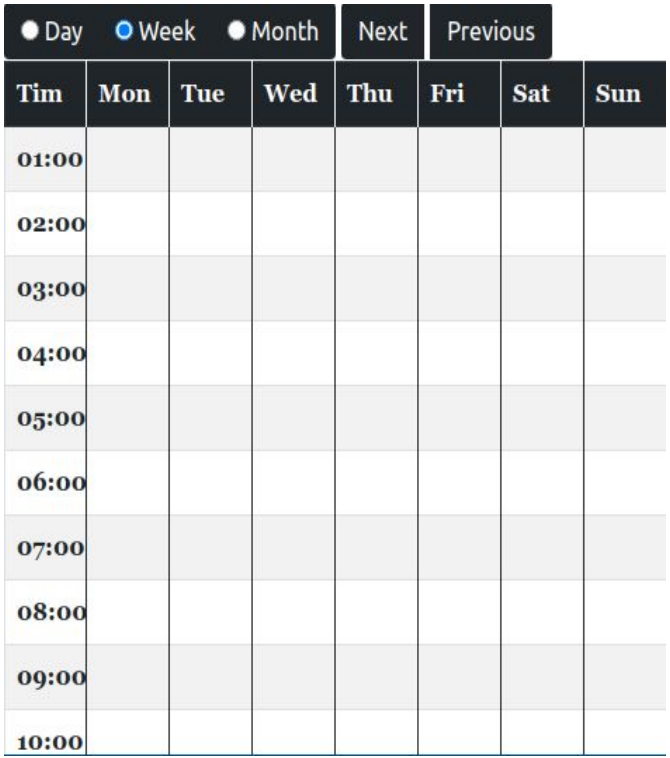
For the first portion of the review we will be looking at three areas pertaining to quality management:

- Usability of the application,
- Testing,
- Quality inspections of the codebase.

Then we will also be covering our risk management.

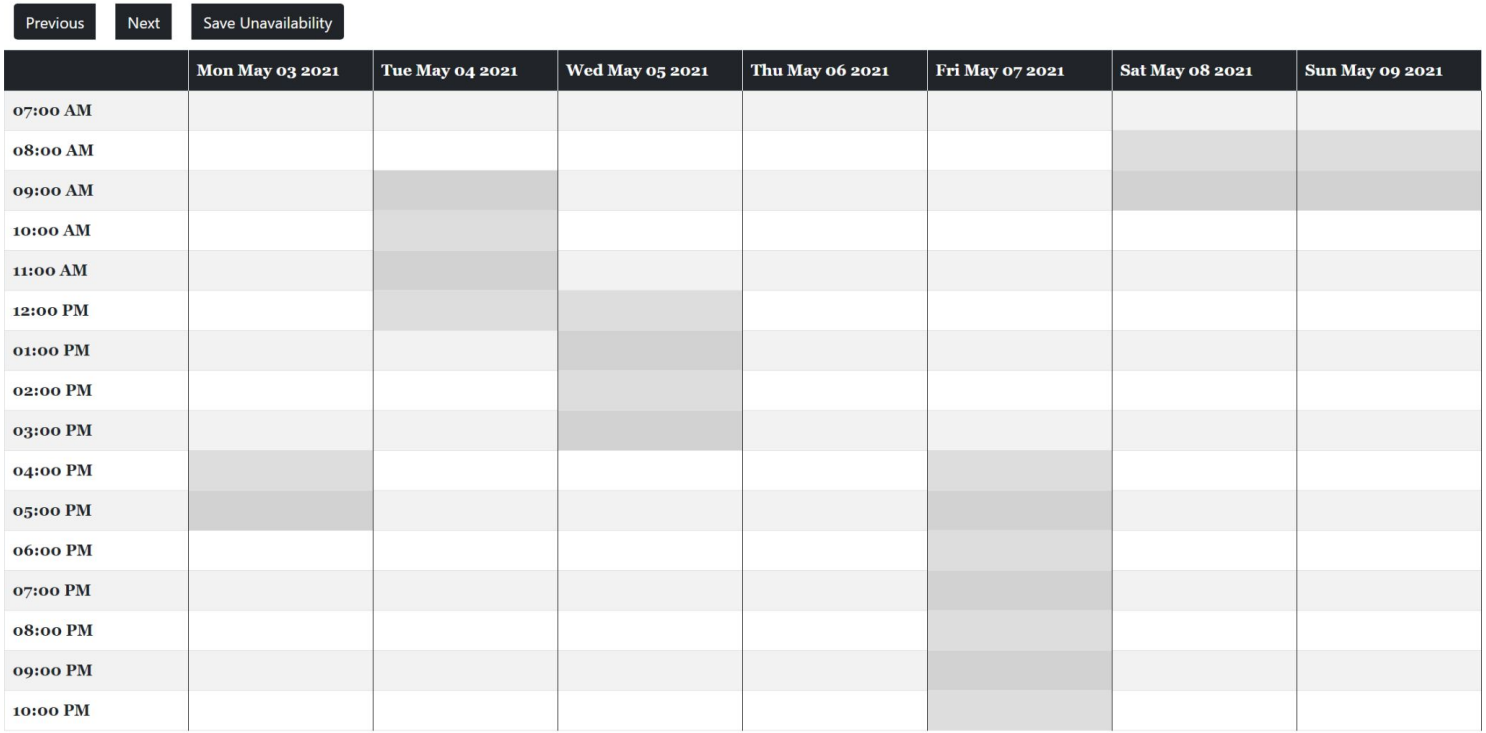
Finally at the end talking about our current Sprint Progress.

Mobile Usability



Usability - Unavailability Selection

Prototype Unavailability Selection

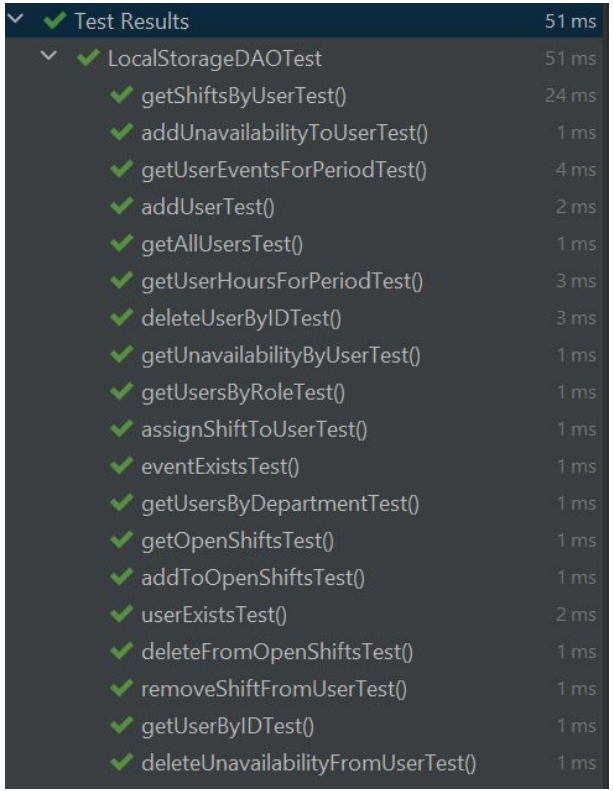


Testing

At the moment most of the computation actually happens in the DataAccessObject.

So this is fully tested in our application using JUnit. Ensuring that we have been passing checks since its implementation in sprint one.

We will be working on more tests for the backend, so we can claim higher coverage.



Code Quality

- Code Standards -
  - We didn’t have set coding standards beside having clear documentation.
  - Code should have clear documentation so that other people know what’s going on.
- Because of the short project timeline and small scope we decided that writing features should be a high priority than high quality code.

- We didn’t run code reviews beside checking over the pages before they were pulled into the master.

- This will heighten the risk of bugs and risk of incorrect story estimates but we feel that the additional features will be worth it.

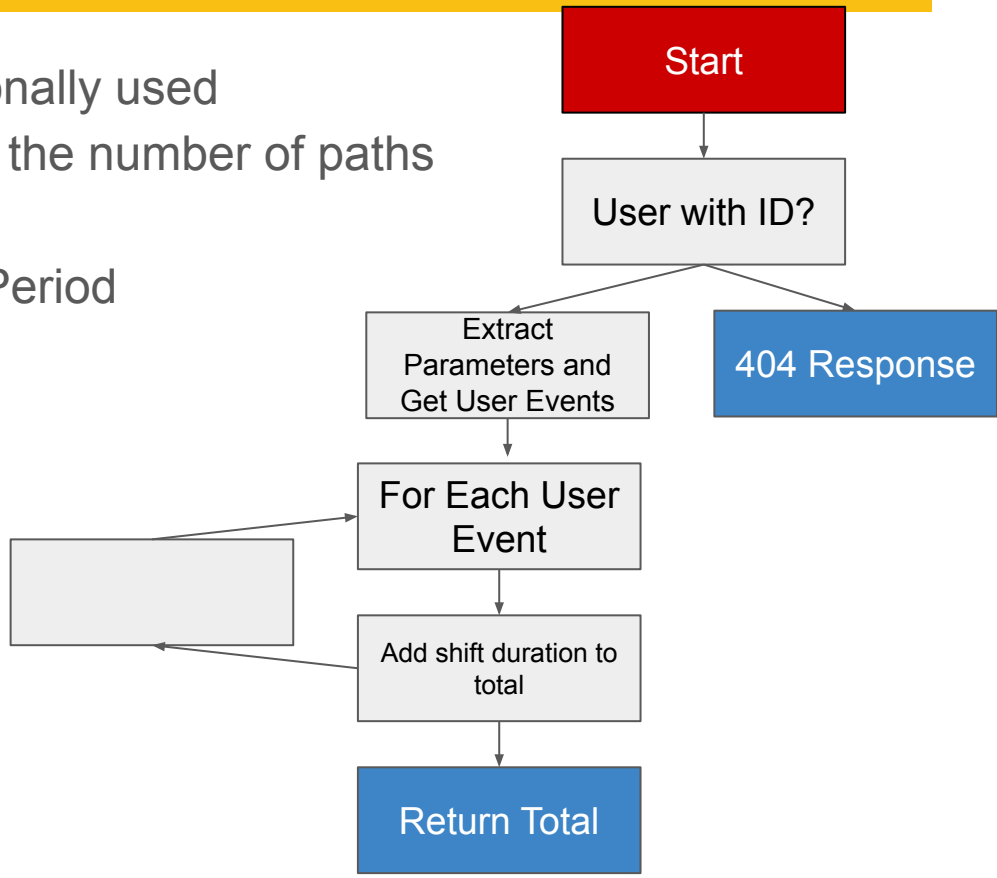
Comment examples

```java
/**
 * This deletes a Unavailability from the specified user.
 * The event is also removed from the event grouping collections.
 * @param userID The ID of the User.
 * @param eventID The EventID of the Unavailability.
 */
public void deleteUnavailabilityFromUser(Integer userID, Integer eventID) {
    User u = usersByID.get(userID);
    Unavailability un = (Unavailability) eventsByID.get(eventID);
    unavailabilityByUser.remove(u, un);
    eventsByDate.remove(un.getStart().toLocalDate(), un);
    eventsByID.remove(eventID);
}
```

```javascript
/* selects the cell that the mouse is in. */
$(".calendar td").on("mouseover", function(){
  $(this).addClass("selected");
});

/* deselects the cell the mouse is in. */
$(".calendar td").on("mouseout", function(){
  $(this).removeClass("selected");
});

/* what happens when next button is clicked */
$("#next-button").on("click", function(){
  alert("Not supported yet!");
});

/* what happens when previous button is clicked */
$("#previous-button").on("click", function(){
```

Cyclomatic Complexity

- While developing, we have occasionally used Cyclomatic Complexity to measure the number of paths used in our methods.
- Example - Getting User Hours for Period
- CC = Edges - Nodes + 2
- CC = 3



Risk Management

Several risks that we have identified:

- Higher than estimated story effort
- Risk of producing bugs
- Different Platform Issues

Risk Reduction Leverage

Risk	Likelihood	Cost (Hours)	Exposure	New risk probability	New Cost (hours)	New Risk Exposure	Risk Reduction Leverage
Different Platform issues	10.00%	12	1.2	5.00%	12	0.6	0.05
Incorrect story estimates	50%	1	0.5	25.00%	1	0.25	0.25
Risk of producing bugs	30.00%	2	0.6	15.00%	2	0.3	0.15

Sprint 2 - Changes

Story ID	Story name	Status	Size	Sprint	Priority
3	Assigning Shifts	Planned	13	2	Must
4	Hours Selection	Planned	13	2	Must
5	View Open Shifts	Planned	8	2	Should
8	Checking availability of employee	Planned	3	2	Must
10	mobile support	Planned	5	2	Must
12	view shifts	Planned	3	2	Must
17	Notifcation system	Planned	13	2	Should
18	Record of hours of each individuals	Planned	3	2	Should
19	Record of total hours	Planned	3	2	Must
23	Filter by roles	Planned	2	2	Should
1	Clocking In&Out	Planned	21	3	Could
6	Request Shift Swap	Planned	21	3	Should
7	Shift Swap Approval	Planned	5	3	Should
11	Recieve shifts	Planned	2	3	Could
13	User guide	Planned	3	3	Could
15	Calendar Export	Planned	13	3	Could

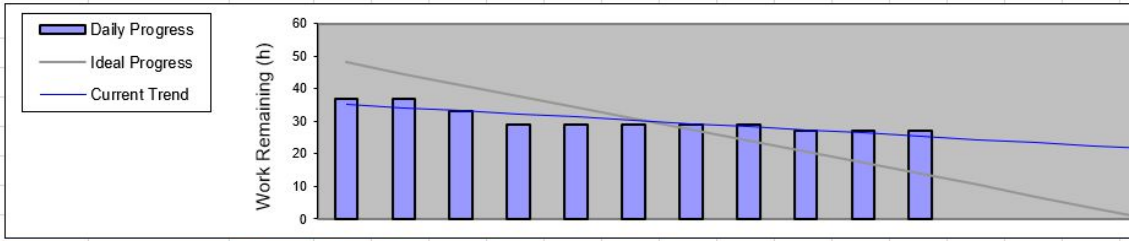
OLD

NEW

Story ID	Story name	Status	Size	Sprint	Priority
3	Assigning Shifts	Planned	13	2	Must
4	Hours Selection	Planned	13	2	Must
8	Checking availability of employee	Planned	3	2	Must
10	mobile support	Planned	5	2	Must
12	view shifts	Planned	3	2	Must
19	Record of total hours	Planned	3	2	Must
27	Site Navigation	Planned	4	2	Must
28	Account Management JS	Planned	4	2	Must
5	View Open Shifts	Planned	8	3	Should
17	Notifcation system	Planned	13	3	Should
18	Record of hours of each individuals	Planned	3	3	Should
23	Filter by roles	Planned	2	3	Should
7	Shift Swap Approval	Planned	5	3	Should
11	Recieve shifts	Planned	2	3	Could
13	User guide	Planned	3	3	Could
15	Calendar Export	Planned	13	3	Could
1	Clocking In&Out	Planned	21		Could

Current Sprint Overview

Sprint 2 Backlog



Sort Sprint Tasks | Update Task Slips

Sprint implementation days	14			Effort	Remaining on implementation day…													
Trend calculated based on last	14	Days	Totals	48	37	37	33	29	29	29	29	29	27	27	27			
Task name	Story ID	Responsible	Status	Est.	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Assigning Shifts	3	Leo	Planned	13	9	9	9	9	9	9	9	9	9	9	9			
Hours Selection	4	Tommy	Planned	13	10	10	6	6	6	6	6	6	6	6	6			
Checking availability of employee	8	Caleb	Planned	3	3	3	3	3	3	3	3	3	3	3	3			
mobile support	10	Leo, Caleb	Planned	5	2	2	2	2	2	2	2	2	0	0	0			
view shifts	12	Keiran	Planned	3	3	3	3	3	3	3	3	3	3	3	3			
Record of total hours	19	Keiran	Planned	3	3	3	3	1	1	1	1	1	1	1	1			
Site Navigation	27	Seu	Planned	4	3	3	3	3	3	3	3	3	3	3	3			
Account Management JS	28	Seu	Planned	4	4	4	4	2	2	2	2	2	2	2	2			

Thank you for your attention.

Are there any questions at this time?