

Documentation

Meeting Minutes

P

Meeting Arrangements

P

Lean Testing

1

P

RESOURCES

SETTING UP HOMESTEAD

13 3 0/20

Dusk on Laravel 5.4

Project Installation

1

Deliverables

Identify User Stories

Define a series of Acceptance Tests

Basic user documentation

Class diagram

1

Database diagram

1

Test Logs

1

User Stories

UC_01: User wants to login to the system

2

P

UC_02: Customer wants to register an account

UC_03: Customer wants to make bookings

P

UC_04: Customer wants to view available slots

P

UC_05: Business owner adds a new employee

P

UC_06: Business owner rosters employees

P

UC_07: Business owner checks booking summaries

P

UC_08: Business owner checks employee's availability for the next 7 days

P

UC_09: Business owner makes booking for a registered customer

P

Product Backlog

Landing Page

User Registration

Dashboard (Customer)

Dashboard (Business Owner)

Database Design

Sprint Backlog

Business Registration

0/5

In Progress

Tests cases new functionality

P

Blocked

Completed (Part B demo)

Paginate availability page

Refactor and Comment code

4/4

Roster issues

Business Owner create bookings for customer using username/email

Customer create bookings

2/2

Implement authentication using innate Auth for security check

4/4

Write new user stories

P

Write project documentation in README file

Gather all required deliverables for submission

Set datepicker for making bookings 1 day ahead of current date

Add working time should check employee's availability

3/3

Add a LOGOUT button on top right

Upload individual logging files to Google Drive

2

P

Completed (before ANZAC)

Booking availability should take into account num_of_slots

2

Add/update staff working days/times later

Missing businessID in
getBookingsByDate in
BookingController

3/3

Redirect to login page whenever try
to access pages without logining

1

P

Add config: default slot period (30
min in this case)

1

Change table bookings: end_time ->
num_of_slots

Add column emp_id to table bookings

Define activities that take different
times

1

Employee no longer has role but the
type of activity in charge

2/2

SOLVE: how to keep track of tables

1

Order booking summary by date

Merging old changes from master to
dev branch

Logging capability

1

A page that allow business owner to
add bookings for customers (using
username/email)

Implemented session

P

Refactored tests

P

Completed (Part A demo)

Integrate business owner functionalities

1

7/7

Integrate customer functionalities

1/1

Implement go back to dashboard button for business owner

Implement showing all workers' availability for the next 7 days (view by business owner)

Present user stories in a PDF

Write up user documentation

P

Set up project from clean install

P

Design class diagram

P

Update Roster Tests cases

P

Design business owner dashboard

Design booking summary page (business owner)

2

Design add new staff page (business owner)

Design view roster page (business owner)

Design add staff working time page (business owner)

4

Design view available slots page/section (customer)

2

Write seeders for employees and rosters tables

2/2

Completed (week 1-6)

Build test units based on user story "Customer wants to register for an account"

1

Build test units based on user story "user wants to login to the system"

P

set up Laravel environment

3/3

P

set up git environment

3/3

P

Implement user registration

3/3

Implement user authentication

Design and implement database for customers, business owners and bookings

3

4/4

Build UI for login/registration page

4

Create brief that denotes user flow between pages

Create prototypes for customer and business dashboards

Mar 24

Add radio buttons for selecting between customer or business accounts on sign in form

Move login error messages to appropriate places

Improve authentication according to channel

Add customer details to booking summary page

Implement Booking related functions

Add seeders needed for demo wk5

3/3

Design customer dashboard

1

Modify user-type to be determined by account name or email

2/2

Design and implement database for employee and its working time/dates

P

Implement adding new employee (by business owner)

Implement adding working time/dates for employee for the next month (by business owner) -> rostering

Fix testing framework. Adopt LeanTesting

P

Tests for Employee related functions

P