ACRS-Team

Private

Part B Tasks

- Refactoring Code
- Datalogging
- Update database schema
- User Stories
- Adding booking types / activities
- Making bookings with a specific employee
- Owner making booking for a customer
- Editing shifts
- Adding business opening/working hours
- GUI Usability

Assignment - Part B - Submission

1. A README file containing all team member names and student numbers your tutor's name and your tute/lab day/time (see under Staff Contacts for tutor name for each tute time)

1/1

2. A PEER ASSESSMENT indicating CONTRIBUTION OF EACH TEAM MEMBER: i.e., what each team member contributed and estimate of overall % contribution (these should add up to 100% for the team). There is a 10% deduction of marks if this is not included.

1 1/1

3. All User Stories, in a single PDF document

2 1/1

4. Short high-level design, e.g., a high level architecture diagram and your class diagram.

1 3/3

5. All source code, appropriately organised into folder hierarchy to match packages etc

1/1

6. A document describing test cases, properly documented (as described in lectures); your submission must also include the status of tests (i.e., passed, being fixed), preferably as a log/screenshot of a management/tracking tool (e.e., leantesting.com)

3 1/1

7. A HOWTO document covering, installing, running your program and User documentation on how to use the webpage for both customer and owner.

1 2 1/1

8. Trello Log

1/1

9. Slack Log

1 1/1

10. Git Link

1/1

11. Minutes of meeting

1 1 1/1

Sprint Backlog Week 8

Refactoring Old Code

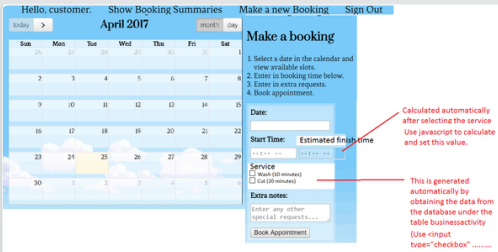
2/2

RT

Editing Pre-existing Work Periods

2 2/2

RT



Adding Booking Types / Activities

1 3 2/2

Extend Bookings with Employees to have no selection for employee and make it look prettier

1/2

S

Update Database Schema

1 1 2/2

Datalogging

2/2

user Input field validation (passwords need to be entered twice, etc.)

2/2

Updating user documentation

3/3

Test Case : Customer make booking with chosen activity

1 1/1

Sprint Backlog Week 7

User Story - Customer & Business owner - Make a Booking

1 1/1

User Stories for Making Booking types

1/1

Make Bookings with Employee

1/1

S

User Story for Making booking with a specific employee

1/1

S

Functionalities for Part B

Some Employees could perform only certain activities

An Employee can be added and their days/times be added or updated later. Employees and their days/times can also be loaded from a configuration file or database.

User Stories - Highest Priority on top.

User Story - Business owner - Make a booking on behalf of the customer

3/3

User Story - Customer - Make a booking

3/3

User Story - Business owner - Editing Working Times for employee

6/6

User story - Business owner - Adding Working Times for employee

1 4/4

User Story - Business owner - Adding new Employee

1 2/2

User Story - Business owner - Show worker availability for next 7 days

2/2

User Story - Business owner - View bookings for next 7 days

2/2

User Story - Business Owner - Show booking summaries

1/1

User Story - Customer - View available appointment times.

1 1/1

User Story - Customer - View booking summary made by the customer

2 1/1

User Story - Business owner - Login

2 3/3

User Story - Customer - Registration

2 2/2

User Story - Customer - Login

2 3/3

Product Backlog

Minutes of meeting log

1

Quality Assurance (QA)

The Scrum Master will assign this list to all completed MVP's (Minimal Viable Product). Ready to be tested for quality assurance. These are the items that need to be cross referenced with existing user stories.

Bug Report

During QA bugs might rise. Use this list to track down all the problems before moving the MVP's (Minimal Viable Product) to done.

Functional Requirements Part A

1. The system should display a main page: When your program starts, a main page should be displayed as login/register pages. The program must authenticate and authorise users and based on user type/role, i.e., business owner or customer.

2. User registration is only for customers; business owner information can be accessible through a file called business.txt (or a db), including business name, business owner name, address, phone, username and password. Customers can register themselves via data entry; this information should be saved in customerinfo.txt (or a db).

3. If the owner can login successfully then the owner is able to add a new employee, add working time/dates for the next month, look at the summaries of bookings, new booking, show all workers' availability for the next 7 days.

4. Customer only view available days/time but not yet book a slot (Part 2);

Assignment - Part 1 - Submission

1. A README file containing all team member names and student numbers your tutor's name and your tute/lab day/time (see under Staff Contacts for tutor name for each tute time)

1 1/1

2. A PEER ASSESSMENT indicating CONTRIBUTION OF EACH TEAM MEMBER: i.e., what each team member contributed and estimate of overall % contribution (these should add up to 100% for the team). There is a 10% deduction of marks if this is not included.

2 1/1

3. All User Stories, in a single PDF document

1 1/1

4. Short high-level design, e.g., a high level architecture diagram and your class diagram.

1 2/2

5. All source code, appropriately organised into folder hierarchy to match packages etc

1/1

6. A document describing test cases, properly documented (as described in lectures); your submission must also include the status of tests (i.e., passed, being fixed), preferably as a log/screenshot of a management/tracking tool (e.e., leantesting.com)

1 1/1

7. A HOWTO document covering, installing, running your program and User documentation on how to use the webpage for both customer and owner.

1 2 2/2

8. Trello Log

1 1/1

9. Slack Log

1 1/1

10. Git Link

1/1

11. Minutes of meeting

1 1 1/1

Sprint Backlog Week 6 - Highest Priority on top.

Implement Functionaility - Showing employee work availability for the next 7 days

1/1

RT

Adding employee shifts

1/1

S

CSS of premade pages

3/3

Written Test Case

5 1

Go through all codes, apply proper commenting/documenting.

1/1

RT

User documentation

1/1

System Diagram

1/1

RT

Documentation on how to install xammp and where to put the repository folder.

1/1

RT

Sprint Backlog Week 5 - Highest Priority on top.

Adding employee

1/1

S

Writing pseudocode tests based on acceptance criteria for all user stories

0/8

Customer - Show available booking slots

1/1

RT

Adding dummy data

2/2

Business owner - View most recent bookings

1/1

RT

Completed Sprint Week 4

Create User Stories for Business booking summaries

1 1/1

S

Implement booking summaries for business

2 1/1

Implement booking calendar

2 1/1

Redoing the Trello board

1/1

S

Update Database Schema

2 1/1

RT

Check currently developed tasks with user stories.

1/1

RT

Customer registration, user must have content in all fields for data validation.

1/1

RT

Completed Sprint Week 3

List of tasks that must be completed during the Scrum sprint.

Calender Page for Booking Page, share LucidChart link

2/2

S

Setting up customer page and owner page

2/2

RT

Create Login and Register page

Mar 18 1 1/1

RT

Completed Sprint Week 2

Connecting calendar and the database

2/2

S

Saturday 18/03 Scrum meeting @ 12pm

RT S

Create script for database tables

2/2

RT

General layout of the page

1 2/2

Site Map & Check QA

1 3/3

Database Schema Design and High level system architecture

2 4 2/2

S

Completed Sprint Week 1

Mock-Up of Login Console

1 2/2

RT

Folder restructure for repository

1 1/1

RT

Read through assignment specifications

Mar 6 4 4/4

  RT S

References / Resources

All files that need to be fetched often. An Invoice template, a scanned document, etc. Important resources added to this list will be accessible with a click.

Scrum Master: Steven. Members: Asli Yoruk, Christine Huynh, Ryan Tran

Assignment 1 Part A Specs, Due 9AM Monday April 10 2017

1

Week1 28Feb: Form your team and send an email about your team members name and student number to your tutor. You also need to setup your collaboration tools, and submit a fake app into the code repository; every member should make at list one submission. You also need to have early team meetings to clarify member roles, and issue tasks for the next week. All meetings must be documented and signed.

Week2 7Mar: Start project documentation, with writing the project scope, class design, and planning work. You also need to start writing the main user stories of this project, e.g., login and/or registration functions. Create mock-up prototypes for these functions. You can define all user management functions validations and limitation after creating their user stories. If you plan to use Test-Driven Development, then you should commence writing the associated automated tests by end of Week 2 / start of Week 3. Otherwise, team members can commence designing Acceptance Tests, based on User Stories as they are written.

Week3 14Mar: In this week, teams should start development of functions that build on the previous week, finish the task, debug, test and submit. You need to also make new tasks based new user stories for new functions such as loading and storing data on users and their selections.

Week4 21Mar: Review previous tasks and fix problems if needed. Develop all assigned tasks for this week, debug, test and submit. Make user stories for new tasks and clarify tasks for the next week. Note: as you write code, you should be adding documentation and logging statements as you go; in particular, logging statements will help you debug your code as it gets complex. You should also be maintaining a bug reporting tool to document / assign / manage bugs.

Week5 28Mar: Review previous tasks and fix problems/bugs if needed. Develop all assigned tasks for this week, debug, test and submit. Write user stories for new functions and clarify tasks for the next week.

Week6 4Apr: Review all previous tasks and fix problems if needed. Develop all assigned tasks for this week, debug, test and submit. Try to polish your code, fix any remaining problems and prepare your full submission, including basic User Documentation. Part 2 specifications will be available and you can start planning new tasks for Part 2.

Week7 11Apr: Submit by 9am Monday; demonstrate your Part 1 submission to your tutor; commence working on Part 2.

Apr 9