|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Activity | Predecessor | Time Estimates | | | Expected |
| Optimal | Normal | Pessimistic |
| A: Registration | - | 3.95 | 4.93 | 5.91 | 4.93 |
| B: Upload | A | 6.58 | 8.22 | 9.86 | 8.22 |
| C: Database | F | 3.29 | 4.11 | 4.93 | 4.11 |
| D: Favourites | B | 1.76 | 2.19 | 2.62 | 2.19 |
| E: Prompt | D | 1.32 | 1.64 | 1.96 | 4.64 |
| F: Pop-up | E | 5.04 | 6.30 | 7.56 | 6.30 |
| G: Ratings | F | 8.12 | 10.14 | 12.16 | 10.14 |
| H: Reliable | I | 3.73 | 4.66 | 5.59 | 4.66 |
| I: High Load | C, G, J | 5.48 | 6.85 | 8.22 | 6.85 |
| J: Flexible | - | 4.83 | 6.03 | 7.23 | 6.03 |

Requirements

A. A registration facility for users and home-owners.

B. Home owners should be able to upload details and pictures of their property onto a central database.

C. The central database should keep track of when a property is available (i.e. when users book a property, the availability dates should be updated accordingly).

D. Whilst browsing, users should be able to keep track of their favourites (stored locally).

E. Once a user has decided upon a property, the website should prompt them to ask them for the dates they require (ensuring that only valid dates are entered). It should also ask them to write a short message introducing themselves to the owner.

F. Once this has been submitted, the owner should be notified and, upon their logging in, should be presented with a pop-up message displaying the details of the request made by the user. They should then either confirm or reject the user’s request.

G. The system should also maintain a ratings-system, whereby home owners and users are given a rating. This rating is updated after the stay is complete – the user rates the home owner and vice versa.

H. The system should be reasonably reliable. A degree of down-time is tolerable, but only very occasionally.

I. The system should be capable of dealing with a reasonably high load of users.

J. As a start-up, it is important that the source code should be reasonably flexible; new requirements are bound to be frequently added.

Dependencies justification

Activity B Depends on A

Home owners are not able to upload details until they have registered a user. This means we need user functionality that is developed from Activity A before we can start on Activity B and link those users to uploading details.

Activity C Depends on F

The database can only keep track of bookings when a user has accepted or finalised a booking which comes from Activity F. Activity F allows the user to accept a booking to make it finalised.

Activity D Depends on B

Activity D which is selecting favourites, cannot be done without having the uploading functionality from B. You will not be able to favourite properties if the properties functionality isn’t created first from Activity B.

Activity E Depends on D

Activity E links to D functionality. D Functionality shows listed properties first, then Activity E is able to select one of those properties to make a requested booking and therefore is a dependency in our development work as well from a Front-end perspective.

Activity F Depends on E

The owner is not able to select a requested booking before a user has request that property to be booked and that is why this dependency has occurred. The development work needs to have Activity F first to add the relevant data and storage before Activity E can query that data.

Activity H Depends on I

Activity H will contain more devOps and Cloud hosting operations and depends on the project to be built fully before it gets deployed onto the cloud for scaling etc. Activity I is basically the last Activity on the development that will optimize code to handle user load. So Activity needs the last development activity to be done before it can package the project ready for deployment.

Activity I Depends on C, G, J

Activity I needs the last of the development requirements to be done before it can optimise the code to handle system user load. We can only optimise when our project functionality has been complete and hence why Activity I depends on C, G, J.