

Development Document

Developer Diary, Chronicon of my Development

You should have the Functional and Non-Functional Requirements:

Versioning:

Might have to come back to this. Should have a discussion of what should be built in **each version**, and **why**.

★ **Version (Core Essentials) 1:**

The version one has to start with your **database**.

The database is started with first, because it is the backbone of everything that is going to be worked on, and I guess the user registration system. (And with this, version 1 is the core essentials).

 **Version (Must Do) 2:** You may wish to bump logging in to this version (okay). You might do the booking, canceling, everything it should do.

 **Version (The Extras) 3:** Altering appoints, branding, color schemes, auditing (unless specifically said), CSS, images...

Minimum of 3 version are needed.

Limitations:

What things in THIS environment (this COLLEGE) is going to impact what I code... say what the limit is (can't use environmental variable, the server, etc) but tell what the limitations is and what the alternatives that were used are.

(Had to hardcore database users into PHP, in real production environment should encode username password into the database, and that adds a layer of security).

(No additional algorithms (so I stick with the password default)).

(Password Hash Lotus Salt)/ \

Version 1:

What do you plan to make?

Document/describe what you made.

What changed and Why did it change.

Drop the FR & NFR's... everywhere possible, because it is good to link.

Evaluate

What will change next version.

Version 2:

What do you plan to make?

Document/describe (print screens, all that) what you made. What changed from the last version and what are you going to do in this version?

What changed and Why did it change.

Drop the FR & NFR's... everywhere possible, because it is good to link.

Evaluate

What will change next version.

At the end of each version, export your database and document it (if you wish to import my database (all that) you can see the data, for proof, I guess, and put it in the folder structure for that version.

Testing

PowerPoint (Impress) of testing evidence and testing table.

Every time you run your code make sure to test it.

Evaluation

Effectiveness of assets and content, make sure the photos are either royalty free or was provided images that by the business (or a copy righter to do stuff).

How appropriate are they. Just make sure to note if they are appropriate, and the legal and ethical implications.

Validity and reliability of the sources used, cite the sources of where you get your stuffy.

Go through your functional and non-functional requirements of the system, and do make sure you evaluate everything so far, and see how many you have got.

They want to see a convincing and perceptive rationale for future iteration is provided (should be a legitimate and thought about process).

Should only be added for contribution, for what you have done.

What do you think about the journey, of what you have done? What have you achieved. (Evaluation of You and Your System.)

First thing in evaluation is the table of functional and nonfunctional requirements, and the KPI and user acceptance criteria, throw the print screens.