SWE30010

HRM PROPOSAL FOR GEMADEPT

Name: Nguyen Dinh Nhat Minh
ID: 103802490
Tutor: Thomas Hang

BACKLOG

This table below shows a portion of our product backlog, specifically just that of sprint one.

No.	Item	Dependencies	Business Value (1 least – 10 most)	Release Schedule
F1	Product UI/UX Design	None	7	Sprint 1
F2	Website for HRM	None	9	Sprint 1
F3	HR database schema design	F2	8	Sprint 1
F4	Manager Portal	F1, F3	7	Sprint 1

1. Sprint Backlog Item to develop

Following discussions with my team members, we have decided to prioritize the development of the ***Website for the HRM Application***. Our decision is based on utilizing the SMART framework, which revealed that our team possesses the most relevant skill sets for this task, including UI/UX design and web coding. Moreover, this will serve as the backbone of our project, enabling us to house further features and functionalities.

The decision not to prioritize Product UX/UI Design was driven by its cosmetic nature, which we determined would not significantly impact the website's functionality. Therefore, it can be deferred to a later stage.

Similarly, the schema design and manager portal will be contingent upon the completion of the website's backbone. As such, they will not be prioritized and will be developed once the website's core functionality is established.

2. Work Breakdown Structure

As a team, we leveraged our collective experiences and expertise to thoroughly discuss and outline the tasks required to complete the sprint backlog items. For the website, we identified five primary tasks that need to be accomplished. These tasks will be further

delineated using the Work Breakdown Structure (WBS) model to ensure a detailed and comprehensive approach to completing this sprint item.

We utilize the Software Development Lifecycle's stages to determine how to break the task down into various sections:

Software Development Lifecycle	Work Breakdown Structure
Planning & Analysis	Requirement Collection
Design	Architecture Design
Implementation	Development
Testing & Integration	Testing & Deployment
Maintenance	No maintenance at this stage as the whole product has not been finalized.

Figure 2.1: SDLC table to determine WBS.

We adopt a hybrid approach, combining both activity-based and product-based methods, to construct our Work Breakdown Structure (WBS). We first concentrate on identifying the main requirements associated with each product, which serves as the foundation for outlining the general framework of our website. Subsequently, we delineate the first task within each product, breaking them down into sub-tasks.

This approach enables us to efficiently address tasks required for specific components within our build, facilitating prompt completion as they arise and ensuring comprehensive control over our product development process.

Through this method, we have devised the design of the WBS diagram as follows:
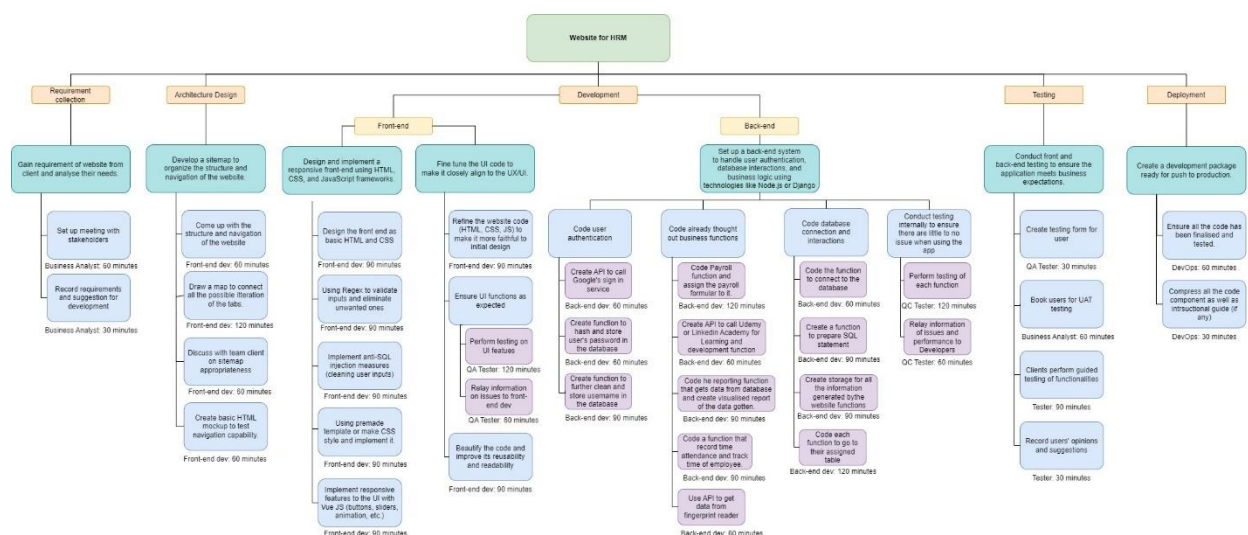

Figure 2.1: WBS Chart with Time Estimation for each task

The WBS structure aids in clarifying tasks by breaking them down into smaller, more manageable components. As a result, each main task that lacks clarity regarding its purpose will be complemented by two to three sub-tasks to provide greater clarity and detail.

3. Rationale and Analysis

The first step is gathering the client's requirements, after which business analysts will schedule a meeting to record the client's website specs. This step requires the least amount of time because it mostly focuses on the client's additional expectations for the website, as the core criteria were specified in the project proposal.

The front-end designer or developer is responsible for drafting the website's layout during the process of building the sitemap. Based on my prior sitemap design experience from my web development degree, this work can be very time-consuming and tedious. It entails laying up every tab and feature that the website might have. Furthermore, extensive team discussions are required to guarantee fluid programming and seamless function linkage; as a result, this task may take longer than expected.

With regard to the principal duty of creating and executing the front-end framework, the design stage may proceed in a gradual manner. That could speed up the process, though, if a UI draft has been created using a platform like Adobe XD or Figma. On the other hand, due to the thorough nature of code development, writing the front-end with the selected framework will require significant resources and effort. Additionally, potential DDOS and SQL injection attacks must be dealt with on the front end, adding another level of complexity and requiring more work.

As a checkpoint to guarantee the promised usability and user-friendliness, the front-end must be modified to match the design precisely once it has been developed with responsive functionality and features. Additionally, testing the website to find and fix any security flaws is part of this step. It goes without saying that in order to fully guarantee security and efficient operation, this stage will also require a substantial time commitment.

As the back-end handles the majority of the website's functions, development inevitably takes a long time. Even though authentication code is time-consuming since it must follow security guidelines, it could nevertheless be completed faster than the website's many features. Creating a database connection is time-consuming because SQL queries must be prepared in order to query the data, and the connection must be kept open at all times so that clients can get the information they need without having to repeatedly shut down. Additionally, because of the nature of the coding needed, coding and building data tables might take a lot of time.

It will take a lot of time and work to carefully review the functions throughout the back-end development stage because QC testing is also necessary to guarantee functionality and fix vulnerabilities.

User testing and feedback gathering comprise the penultimate step, which can be finished really rapidly. The main task of this step is to arrange user testing sessions and gather input to improve the website. Other than scheduling users' spare time, it doesn't require any labor.

DevOps must be included in the deployment stage of the process in order to examine all pertinent product details prior to packaging the product for production. DevOps shouldn't have too much trouble determining the product's readiness if all testing proceeds as planned, making this task reasonably rapid and not unduly time-consuming.