# Final Functional System-Sprint 1 [70 Points, Total Weight: 20%]

# Due date:

In this last sprint, you will finalize the implementation of the system. The delivery of this sprint is the complete system that will be presented to the external examiners.

|  |  |  |  |
| --- | --- | --- | --- |
| **Sprint #** | **Points** | **Description** | **Duration** |
| Sprint 1 | 20 | System implementation | 2 weeks from the start date |
| Sprint 2 | 5 | System implementation | 2 weeks from the start date |
| Sprint 3 | 5 | System implementation | 2 weeks from the start date |
| Sprint 4 |  | Testing and final delivery |  |

# Grading Rubric

|  |  |  |
| --- | --- | --- |
| **Criterion** | **Tasks** | **Points** |
| **Documentation** | * The code is well commented and it is also self-describing * Use of summary to document methods * Do not document properties unless they are not clear * Each created file starts with a consistent header comment * The application has Readme file on GitHub that describes the application * Every possible scenario of interaction with the system was considered (not only success scenario but also the alternative ones). | 10 |
| **Coding Standards** | * Code is well-organized, understandable and readable * Validation and display annotations are added to properties. * Use of validations and restricted length textboxes that match the database attribute length/size. * C# coding style is followed (e.g. Pascal/Camel notation) * Variety of Master pages with corresponding content pages per different users * Exception handling is used especially when inserting data into the database to avoid thrown unhandled exceptions on the screen. * Sessions are used to exchange objects between pages and some forms are auto-filled based on retrieved data from the database | 5 |
| **Sprint Document** | * Sprint overview is present with at least 100 words * Sprint details include use cases, short descriptions, and user interfaces to be implemented (see GUI document) * All uses cases and their percentage of implementation are stated | 5 |

|  |  |  |
| --- | --- | --- |
| **Delivery** | * Application is delivered on time and on GitHub * 30% of the total use cases stated in the project report are implemented * Each use case implemented is more than 90% complete * Database is seeded with enough examples to test the use cases implemented. | 10 |
| **User Interface** | * All necessary properties are present in the view * User interface is easy to navigate and consistent * User interface is organized and user friendly. * Colors, fonts, spacing are consistent throughout the app * Implemented user interface is 70% similar to the prototyped user interface (see UI design document) * Validation messages are user friendly * Placeholders are added to help user input correct data * Application is easy to navigate using menus and links | 10 |
| **Runtime** | * Application runs without warnings and errors * Solution is efficient and clear to understand * Code is reused whenever it is possible * Notification messages shown on insertion or editing of information as well as any error occurring. | 20 |
| **Sprint Manual** | * Sprint screenshots and navigation menus * Demonstration of screen usage * Demonstration of user input validation | 10 |

1. **Sprint Document Template**
   1. **Sprint Overview**
      * Discuss briefly the objectives of this sprint
      * What the user of your application should be able to do by the end of this sprint
      * This section should be approximately 100 words

# Sprint Completeness

* + - List use cases numbers, with short descriptions, implemented in this sprint
    - List the user interfaces with their numbers that are implemented in this print

|  |  |  |  |
| --- | --- | --- | --- |
| **UI** | **Use Case** | **Use Case Title** | **Completed (%)** |
|  |  |  |  |
| 1 | 1 | Create New Employee | 100 |

**3. Sprint Manual**

* Look up the attached manual.
* Put screenshots of your sprint 1.
* Put a short description below each screenshot.

1. **Sprint Retrospective on GitHub**

What went well?

* + 1. List the users cases and the user interfaces that you implemented successfully in this sprint. You may write in narrative or list style.

What did not go well?

* + 1. Discuss the user cases and tasks that were not implement or not fully implemented. Provide the reasons for these issues.

How should you improve?

* + 1. Indicate at least two ways to improve your work in the next sprint and be as detailed as you can.

# Deliverables

1. Submit your **final application** on your GitHub repository with a Readme file describing the application. Create a branch named Sprint 1 for this submission.
2. Submit the link to your GitHub repository where your application is uploaded. Include your link on Blackboard.
3. Submit the link to your GitHub Pages where your API documentation is located. Include your link on Blackboard.
4. Submit the sprint-planning document as PDF named Firstname\_Sprint1.pdf. Submit a soft copy to Blackboard.