**Main Project**

Complete a Project that has a socially beneficial outcome. Identify Real projects - Any project useful to the Society. The project must be done inhouse. The student must spend the time in the lab for project work.

Attendance as per MCA regulations is applicable for appearing in the final viva-voce. However, the evaluation committee can give consent to students in exceptional cases to do their project in Industry which has real live projects. Local industries and training Institutes which offer live projects should not be permitted.

Students individually must do a project approved by their faculty Supervisor. Project evaluation weights shall be as follows: -

For convenience the marks are allotted as follows.

* Project Progress evaluation details
* Total Marks for the Final Project: 100
* Project evaluation by the supervisor/s : 30 Marks
* Presentation & evaluation by the Committee: 40 Marks
* Evaluation by the External expert: 30 Marks

The project assessment board shall consist of the following members.

* Chairman: Head of the Department
* Members: Project supervisor/s of the student
* One faculty member from the Department
* One faculty member from a sister Department
* An external expert, either from an academic/research institute or Industry

A faculty/technical staff should act as the Scrum Master of each Project team. The Customer or a Senior faculty is the Product Owner.

Frequent meetings are highly encouraged, at the convenience of the Scrum Master. Should not exceed 15 minutes. Ensure meetings once in three days. A sprint is two weeks, so ensure biweekly reviews. A review should not exceed 30 minutes. A demo to the Product Owner is compulsory in each review.

Use git for Version control.

Follow Test Driven Development. Bugzilla or an equivalent tool may be used for bug tracking. The student should keep a rough record. Divide it into 4 parts. Product Backlog, Database & UI Design, Testing & Validation and details of Versions. Make dated entries to the corresponding part, as the project progresses. The Corrections and comments from Product Owner/Scrum Master should be clearly indicated with the Date.

Project presentations may be conducted for Internal Assessment. They should also serve as supplement to Scrum reviews. The evaluation board may consist of other faculty members/technical staff. A maximum of 2 Presentations are allowed. Scrum reviews should not be sacrificed for presentations.

Students must be encouraged to publish their work in journals and due credit to be given to the students for this.

Latex or an equivalent tool should be used for preparing Presentations and Project Report.

**Project Course Plan**

|  |  |  |
| --- | --- | --- |
| **Week** | **Dates** | **Schedule** |
| 1 | 28 January to 1 February 2019 | * Selection of Topic, Feasibility analysis. |
| 2 | 4 February 2019 | * Topic Submission |
| 2 | 5 February 2019 | * Topic Approval, Meeting of student and Scrum Master with Product Owner. |
| 2 | 6 February 2019 | * Informal, preliminary discussions of requirements.   Creating user stories in the rough record. |
| 2 | 6 February 2019 | * User Story discussion with Scrum Master * GIT Repository creation |
| 3 | 11 February 2019 | * Identifying modules, Initial Design of Database & UI. * Pushing the first version of the Project along with a Readme file containing contact details of team members. |
| 3 | 14 February | * Using Branch for individual members. * Merging with Master. |
| 4 | 18 and 19 February 2019 | * First Scrum Review. (Here onwards, the Scrum reviews are conducted on every other week) |
| 7 | 11 March to 15 March 2019 | * Project Presentation - Interim Evaluation to be based on Git History |
| 12 | 25 March to 30 March 2019 | * Project Presentation - Final Evaluation to be based on Git History |
| 13 | 4 April 2015 | * Submission of Project Report, with Rough Record |

**Course Outcomes**

1. Identify problems that can be solved using computer-based solutions.
2. Design and build software applications that can solve problems.
3. Follow best practices and ethical principles when designing software systems.
4. Follow agile software development principles to build software systems.
5. Use tools and packages to make development process easier and simpler.
6. Prepare project reports and communicate effectively the result of one’s work. Adherence

**Course Outcomes mapping**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 |
| CO1 | 1 | 2 |  | 2 |  |  | 1 |  |  |  |
| CO2 | 1 |  | 2 | 1 |  |  |  |  |  |  |
| CO3 |  |  |  |  |  | 2 |  |  |  |  |
| CO4 | 1 |  |  |  |  |  | 1 |  |  |  |
| CO5 |  |  |  |  | 2 |  |  | 1 |  |  |
| CO6 |  |  |  |  |  |  |  |  | 2 |  |

**Course Outcomes mapping**

|  |  |  |  |
| --- | --- | --- | --- |
|  | PSO1 | PSO2 | PSO3 |
| CO1 | 1 |  | 2 |
| CO2 | 1 | 2 |  |
| CO3 | 1 | 2 |  |
| CO4 | 1 | 2 |  |
| CO5 |  | 2 |  |
| CO6 |  |  | 1 |