## 

| **International School**  **CAPSTONE PROJECT 1**  **CMU-SE-450 / CMU-IS-450 / CMU-CS-450**  **PROJECT PLAN**  Version 1.1  Date: 12 - Aug - 2020  **SMART DASHBOARD APPLICATION**  **Submitted by**  Vo Van Hoa  Pham Van Tin  Ky Huu Dong  Tran Thanh Kieu  **Approved by**  **Capstone Project 1 - Mentor:**  Name Signature Date  Binh, Thanh Nguyen \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_14 - Dec- 2020    Name Signature Date  Huy, Truong Dinh \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| --- |

| **PROJECT INFORMATION** | | | |
| --- | --- | --- | --- |
| **Project Acronym** | SDA | | |
| **Project Title** | Smart Dashboard Application | | |
| **Project Web URL** | <https://sda-research.ml/> | | |
| **Start Date** | 12 - Aug - 2020 | | |
| **End Date:** | 07 - Dec - 2020 | | |
| **Lead Institution** | International School, Duy Tan University | | |
| **Project Mentor** | PhD Binh, Nguyen Thanh; MSc Huy, Truong Dinh | | |
| **Scrum Master** | Hoa, Vo | hoavo.dng@gmail.com | 0935.193.182 |
| **Team Members** | Tin, Pham Van | tinphamvan123@gmail.com | 0932.535.175 |
| Dong, Ky Huu | kyhuudong@gmail.com | 0898.246.980 |
| Kieu, Tran Thanh | thanhkieutran391@gmail.com | 0358.583.251 |

| **DOCUMENT INFORMATION** | | | |
| --- | --- | --- | --- |
| **Document Title** | Project Plan | | |
| **Author(s)** | Team C1SE.06 | | |
| **Role** | [SDA] Project Plan\_v1.1 | | |
| **Date** | 16 - Nov - 2020 | File name | [SDA] Project Plan\_v1.1 |
| **URL** | <https://github.com/sdateamdtu2020/sda-documents> | | |
| **Access** | Project and CMU Program | | |

## REVISION HISTORY

| **Version** | **Person(s)** | **Date** | **Description** |
| --- | --- | --- | --- |
| Draft | Hoa, Vo | 31 - Aug - 2020 | Initiate document |
| 1.0 | All members | 22 - Sep - 2020 | Finish content of document |
| 1.1 | Tin, Pham | 16 - Nov - 2020 | Update task schedule |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# 

# 

# **TABLE OF CONTENTS**

PROJECT INFORMATION 1

DOCUMENT INFORMATION 1

REVISION HISTORY 2

**TABLE OF CONTENTS 4**

[**I. INTRODUCTION**](#_2t80i2oud222) **5**

[1.1 PROJECT NAME](#_1fob9te) 5

[1.2 PROJECT DURATION](#_9mtbazhszp8f) 5

[1.3 GOAL OF PROJECT](#_7457npcq3fce) 5

[1.4 PROJECT SCOPE](#_fyzkse89o7p0) 5

[**II. TEAM ORGANIZE**](#_kk0wm2e7aliy) **6**

[2.1 SCRUM TEAM INFORMATION](#_6b6q3crun518) 6

[2.2 ROLES AND RESPONSIBILITY](#_hx7qlx4j06or) 6

[**III. SCHEDULES**](#_k6rksj500e8j) **7**

[3.1 OVERALL SCHEDULES](#_56cuucb2rk2h) 7

[3.2 DETAIL SCHEDULES](#_yd7szvyycbij) 7

**IV. COST 9**

[4.1 RESOURCES](#_9wt9nfy5jltd) 9

4.2 TOTAL COST ESTIMATE 9

[**V. DEVELOPMENT PROCESS**](#_isdw52oscg74) **9**

[**VI. DEVELOPMENT ENVIRONMENT**](#_m37kb6b8vjrp) **12**

[**VII. COMMUNICATION & REPORTING**](#_wzohjr8dhpf) **1**3

[7.1 REPORTING METHODOLOGY](#_fhko2vjzyza4) 13

[7.2 COMMUNICATION METHODOLOGY](#_jtorc5vudr43) 13

**VIII**[**. RISKS**](#_n2ocvgyjics8) **1**4

**I**[**X. DELIVERABLES**](#_gzvrncdluxt4) **1**5

[**X. CONFIGURATION MANAGEMENT**](#_pxuzc1jdauxu) **1**6

**XI. REFERENCE 16**

# 

# **INTRODUCTION**

## PROJECT NAME

The project’s name is: **“Smart Dashboard Application”**

Team members:

1. Hoa, Vo Van Scrum Master
2. Tin, Pham Van Team Member
3. Dong, Ky Huu Team Member
4. Kieu, Tran Thanh Team Member

## PROJECT DURATION

* Project will be started on: 12 - Aug - 2020
* Project will be finished on: 14 - Dec - 2020

## GOAL OF PROJECT

Smart Dashboard Application - SDA - means that a dashboard that is convenient for users to analyze and review data. It will include several hundred datasets about real-time information of the environment, historical aerial photographs, measurement data of air pollutants… SDA will connect and analyze data from multiple sources in ways you’ve never imagined. Then reveal the insights you’ve been missing…in just a matter of minutes.

With smart suggestions and an intuitive visual interface, SDA makes it easy for any user to combine data and discover hidden insights in one place…without the usual scripting, coding, and IT hand-holding. With this dashboard, individuals or any subjects can take advantage of environmental data to be able to decide the best relevant policies.

## PROJECT SCOPE

In this project scope, we just implement these features about Forest, Atmosphere, Climate, Population, Industry

* View list of data cubes (DCs)
* Select 1 DC, drag to active page of SDB
* Visualise active DC in different type (charts, maps, data tables)
* Select a new DC to active page
* Link 2 DCs in SDB
* Creating new DC based on the 2 linked DCs
* Visualize new DC
* Continue with increase active DCs

Language:

* English
* Vietnamese

Duration: 17 weeks.

# **TEAM ORGANIZE**

## SCRUM TEAM INFORMATION

| **Full Name** | **Email** | **Phone number** | **Role** |
| --- | --- | --- | --- |
| Hoa, Vo | hoavo.dng@gmail.com | 0935.193.182 | Scrum master |
| Tin, Pham Van | tinphamvan123@gmail.com | 0932.535.175 | Team member |
| Dong, Ky Huu | kyhuudong@gmail.com | 0898.246.980 | Team member |
| Kieu, Tran Thanh | thanhkieutran391@gmail.com | 0358.583.251 | Team member |

## ROLES AND RESPONSIBILITY

| **Role** | **Responsibility** | **Name** |
| --- | --- | --- |
| **Product Owner** | Understand the user and customers with their needs.  Collaborate with the development team.  Manage the stakeholders.  Describe the user experience and product features.  Provides detailed user stories. | PhD Binh Thanh Nguyen |
| **Scrum Master** | Communicate the value of Scrum  Teach the organization on Scrum to maximize business value  Facilitate Sprint Planning, Daily Scrums, Sprint Reviews and Retrospective Meetings  Create the Task Board and Sprint Burndown Chart at the start of every Sprint  Attend all Scrum meetings  Preserve the integrity and spirit of the Scrum framework  Maintain the focus of the Team  Make the Team aware of impediments and facilitate efforts to resolve them  Serve as a coach and mentor to members of the Team  Respectfully hold the Team, Product Owner and Stakeholders accountable for their commitments  Continually work with the Team and business to find and implement improvements | Hoa, Vo Van |
| **Secretary** | Record the content of group meetings and activities of the member | Kieu, Tran Thanh |
| **Reviewer** | Analysis of the functions and requirements of the product.  Review documents related to the project | Tin,Pham Van |
| **Team member** | Frontend Developer: Dashboard | Tin,Pham Van |
| **Team member** | Back-end Dev: RDF Data Cubes Dev | Hoa, Vo |
| **Team member** | Back-end Dev: Database Dev | Dong, Ky Huu |
| **Team member** | Front-end Dev: Homepage  Tester, Secretary | Kieu, Thanh Tran |

# **SCHEDULES**

## OVERALL SCHEDULES

| **No** | **Phase** | **Iteration** | **Start Day** | **End Day** |
| --- | --- | --- | --- | --- |
| 1 | Development | SPR1 | Aug 12, 2020 | Sep 10, 2020 |
| SPR2 | Sep 11, 2020 | Oct 10, 2020 |
| SPR3 | Oct 11, 2020 | Nov 9, 2020 |
| SPR4 | Nov 10, 2020 | Dec 12, 2020 |

### 

## DETAIL SCHEDULES

| **WBS** | **Task** | **Task owner** | **Start** | **End** | **Days** |
| --- | --- | --- | --- | --- | --- |
|  | **Sprint 1** |  | **Aug 12, 2020** | **Sep 10, 2020** | **30** |
| 1 | Data Modeling | Dong, Hoa |  |  |  |
| 2 | Physical Database Design | Hoa |  |  |  |
| 3 | Set up database | Dong |  |  |  |
| 4 | Build UI | Tin, Kieu |  |  |  |
| 5 | Testing | Kieu |  |  |  |
|  | **Sprint 2** |  | **Sep 11, 2020** | **Oct 10, 2020** | **30** |
| 1 | Finding Data | Dong |  |  |  |
| 2 | Data processing | Dong |  |  |  |
| 3 | Data Warehouse Designing | Hoa |  |  |  |
| 4 | ETL Process Validating | Hoa |  |  |  |
| 5 | Building UI | Tin |  |  |  |
| 6 | Testing | Kieu |  |  |  |
|  | **Sprint 3** |  | **Oct 11, 2020** | **Nov 9, 2020** | **30** |
| 1 | Expanding database | Dong |  |  |  |
| 2 | Crawl data | Dong |  |  |  |
| 3 | RDF Data Cubes Designing | Hoa |  |  |  |
| 4 | Setting up DW to RDF Process Tool & Environment | Hoa |  |  |  |
| 5 | Implementing DW2RDF Process | Hoa |  |  |  |
| 6 | Setting up RDF Data Cubes Storing & SPARQL Endpoint | Hoa |  |  |  |
| 7 | Building UI | Tin |  |  |  |
| 8 | Testing | Kieu |  |  |  |
|  | **Sprint 4** |  | **Nov 10, 2020** | **Dec 12, 2020** | **30** |
| 1 | Documents | All members |  |  |  |
| 2 | Building a history data source | Dong |  |  |  |
| 3 | Validating RDF Data Cubes | Dong, Hoa |  |  |  |
| 4 | Building SPARQL-REST API | Hoa |  |  |  |
| 5 | Building UI | Tin |  |  |  |
| 6 | Integrate | Hoa, Tin |  |  |  |
| 7 | Testing | All members |  |  |  |
| 8 | Deploy | All members |  |  |  |
| 9 | Release | All members |  |  |  |

# COST

## RESOURCES

| **Full Name** | **Role** | **Salary Rate (USD / Hour)** |
| --- | --- | --- |
| Hoa Vo | Scrum Master | 2.0 |
| Tin Pham | Team Member | 2.0 |
| Dong Ky | Team Member | 2.0 |
| Thanh Kieu | Team member | 2.0 |

## TOTAL COST ESTIMATE

| **Sprint** | **Duration(hours)** | **Cost ($)** |
| --- | --- | --- |
| 1 | 376 | $752 |
| 2 | 356 | $712 |
| 3 | 387 | $774 |
| 4 | 498 | $996 |
| **Total** | **1617** | **$3234** |

# **DEVELOPMENT PROCESS**

**PRINCIPLE AND DIFFERENT STAGES**

The SCRUM methodology relies on the incremental development of a software application while maintaining a completely transparent list of upgrade or correction demands to be implemented (backlog). It involves frequent deliveries, usually every four weeks, and the client receives a perfectly operational application that includes more and more features every time. This is why the method relies on iterative developments at a constant rhythm of 2-4 weeks. Upgrades can therefore be more easily integrated than when using a V-cycle.

This method requires four types of meetings:

* Daily meetings: the entire team meets for approximately 15 minutes every day in order to answer the following three questions, usually while standing: what did I do yesterday? What am I going to do today? Is there a cumbersome impediment today?
* Planning meetings: the entire team gathers to decide on the features that will make up the following sprint
* Work review meetings: during this meeting, every member presents what he has done during the sprint. They organize a demonstration of the new features or a presentation of the architecture. This is an informal meeting lasting for approximately 2 hours which is attended by the entire team.
* Retrospective meetings: at the end of each sprint, the team analyzes both successful and unsuccessful elements of their activity. During this meeting lasting between 15 and 30 minutes where everyone is invited and speaks on their own behalf, a vote of confidence is organized in order to decide on the improvements to be made.

The advantage of this method consists in reducing the documentation to the minimum in order to gain productivity. The idea is to write only the minimum documentation which allows to save the history of the decisions taken on the project and to easily perform interventions on the software when it goes into the maintenance phase.

**AGILE - SCRUM ORGANISATION**

The SCRUM methodology involves the following three main players:

* Product owner: In most projects, the product owner is the leader of the client's project team. He is the one who will define and prioritize the product features and choose the date and content of each sprint based on values (workloads) that the team communicates to him.
* Scrum Master: He is a genuine facilitator on the project as he makes sure that everyone works at their full potential by eliminating impediments and protecting the team from external interference. Moreover, he pays particular attention to the respect of the different SCRUM phases.
* Team: A team is typically made up of 4-10 people and groups together all the IT specialists who are necessary on a project, i.e. an architect, a designer, a developer, a tester etc. The team is self-organizing and remains unchanged during an entire sprint.

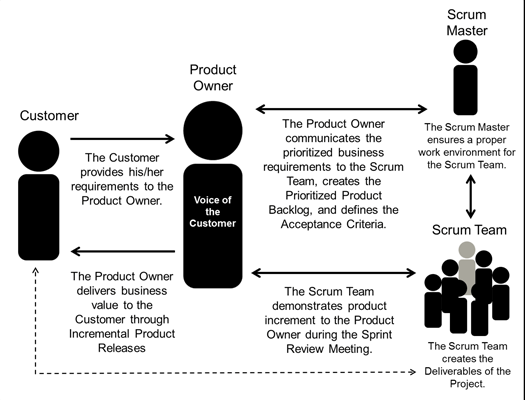


Figure 4.2: Scrum team members

**AGILE - SCRUM ADVANTAGES**

Scrum differs from other development methods through its advantages which turn it into a pragmatic response to product owners' current needs:

Iterative and incremental method: this allows to avoid the "tunnel effect", i.e. the fact of seeing the result only at the final delivery, and nothing or almost nothing during the entire development phase, which is so frequent with V-cycle developments.

Maximum adaptability for product and application development: the sequential composition of the sprint content allows to add a modification or a feature which was not initially planned. This is precisely what renders this method "agile".

* Participatory method: every team member is asked to express his opinions and can contribute to all the decisions taken on the project. He is therefore more involved and motivated.
* Enhancing communication: by working in the same development room or being connected through different communication means, the team can easily communicate and exchange opinions on the impediments in order to eliminate them as early as possible.
* Maximizing cooperation: daily communication between the client and the team enables them to collaborate more closely.
* Increasing productivity: as it removes certain "constraints" of the classical methods, such as documentation or exaggerated formalization, SCRUM allows for increased team productivity. By adding to this the qualification of each module which allows to determine an estimation, everyone can compare their performance to the average team productivity.

**RISKS AND SOLUTIONS**

The SCRUM methodology does not offer a universal answer to all the problems that are inherent to software development. Teams need to pay attention to the risks below, which, nevertheless, offer a systematic answer based on extrapolating the method:

* Team size: team size is typically limited to 7 or 10 people and can become an impediment if it exceeds these recommendations. In the latter case, the organization of meetings becomes impossible and the very foundations of the method are affected. The solution is to set up a Scrum of Scrums. This consists in dividing the project into teams of appropriate sizes and adding an instance of a higher level which groups together the Scrum Master of each Scrum.
* Multiple requests: Requests may be transmitted through several channels on a project and can sometimes be difficult to manage due to their contradictory aspects. These contradictions can slow down the delivery validation process. In order to solve this problem, it is vital to use a single request management tool, which is a standard option with projects.
* Development quality: The more the number of teams increases, the more difficult it becomes to deal with quality. This rule is all the more true when the project is distributed among several centers. The main risks are related to code quality and the number of bugs identified during integration. This is why it is important to have a rigorous quality policy and a project quality plan which precisely defines the rules of the project. Frequent code audits and the implementation of indicators that measure the developers' performance allow minimizing this risk.

# **DEVELOPMENT ENVIRONMENT**

| **Component** | **Development Environment** |
| --- | --- |
| **Operating system** | * Windows * Linux |
| **Development Tools** | * Visual Studio Code * Pgadmin4 * Fluent Editor |
| **Data Warehouses** | * PostgreSQL |
| **Data Cubes** | * RDF Graph * OpenRefine * CubiQL * Jena |
| **Client dependencies** | * NodeJS: 14.5 * NPM: 6+ * React: 16.13.1 * Redux |
| **Server dependencies** | * ExpressJS * Node-Postgres |
| **Third-party dependencies** | * Source Code Version Control: [Github](https://github.com/sdateamdtu2020). * Gcloud services: Compute Engine, Firebase. * Server Endpoint: GraphDB * Material-UI, react-dnd, redux-toolkit, highcharts,... |

# **COMMUNICATION & REPORTING**

## REPORTING METHODOLOGY

| **Audience/ Attendees** | **Topic/ Deliverable** | **Frequency** | **Method** |
| --- | --- | --- | --- |
| - Product Owner  - Scrum Master  - Team Members | Project Progress Review | Weekly | Email, Skype or Zoom |
| - Product Owner  - Scrum Master  - Team Members | Explicit Requirement | When needed | Email, Skype or Zoom |
| - Mentor  - Scrum master  - Team members | Milestone review | End of each Milestone | Skype, Zoom |
| - Scrum master  - Team members | Daily tasks | Each day | Slack, Discord, Messenger |

## COMMUNICATION METHODOLOGY

| **Type of**  **Communication** | **Communication**  **Schedule** | **Communication**  **way** | **Who Initiates** | **Recipient** |
| --- | --- | --- | --- | --- |
| Status Report  ( Daily meeting) | Daily | Slack, Discord | Scrum Master | Scrum Team |
| Schedule and  Effort Tracking | Daily | Face to face or GSheets | Scrum Master | Scrum Team |
| Work Review | Daily | Face to face | Scrum Master | Scrum Team |
| Work Report | Every Wednesday, Sunday | Face to face or Discord | Scrum Master | Scrum Team |
| Project Review,  ask problems | Every Wednesday, Sunday  (flexible) | Face to face, Discord or Zoom | Scrum Master | Scrum Team,  Mentor |
| Ask & Review  problems | Anytime | Face to face, Slack, Discord, Zoom | Scrum's Member | Mentor and  Scrum Team |

# **RISKS**

In this part of the document, it contains several risks that could happen to the development team in the future. It also includes probability, severity and mitigation strategy for each risk.

| **Risk** | **Definition** | **Probability** | **Severity** | **Mitigation Strategy** |
| --- | --- | --- | --- | --- |
| Lack of coding experiences | No one in team member work with Python, React, Express, Data Warehouses, Data Cubes, Crawl data | H | M | Each team member has to learn and help the other to learn quickly. |
| Source Code conflict | Problems while merging code between members to master branch | H | H | Each team member must resolve conflicts by using git merge CLI before merging to master branch. |
| Member conflict | Team member maybe conflict with each other while discussing | H | M | Team building, playing board games to get everyone together. |
| Less equipment | No machine or hosting for deploying the server. | L | M | Try free hosting for deployment. |
| Time management | Every member has to go to work or school. | H | H | Overtime |
| Language barrier | Most of documents the are in English, sometimes it hard to understand clearly the articles and the informations | H | H | Improve the individual English skills in meanwhile doing the project. Asking the mentor technology for specific |

| **Probability** | | **Severity** | |
| --- | --- | --- | --- |
| L | Rarely happened. | L | Low damaged |
| M | Sometime happened | M | Medium damaged |
| H | Usually happened | H | Serious damaged |

# **DELIVERABLES**

| **No** | **Activities** | **Deliverables** |
| --- | --- | --- |
| 1 | Project Proposal | Project Proposal Document 1.2 |
| 2 | Project Plan | Project Plan Document 1.1 |
| 3 | Product Backlog | Product Backlog Document v1.1 |
| 4 | Architecture Document | Architecture Document v1.2 |
| 5 | Database Design | Database Design Document v1.2 |
| 6 | Interface Design | Interface Design Document v1.2 |
| 7 | Test Plan | Test Plan Document v.1.2 |
| 8 | Test Case | Test Case Document v1.3.1 |
| 9 | Acceptance Criteria | Acceptance Criteria v1.0 |
| 10 | Sprint Backlog & Burndown Chart | Sprint Backlog & Burndown Chart v1.4 |
| 11 | Team Reflection | Team Reflection v1.0 |
| 12 | Technologies Stack | Technologies Stack Document v1.2 |

# **CONFIGURATION MANAGEMENT**

| **No** | **Tool** | **Content** |
| --- | --- | --- |
| 1 | Google Sheet | Track member activities. At the end of each day, team members will post on time log and scrum master will check. |
| 2 | Google Document | Track the changing of documents & manage versions of documents. |
| 3 | Github | Repositories for source code version management. |
| 4 | Weekly Meeting | Hold a meeting every week to assign tasks to each member.  If there are some emergencies but we cannot sit together then we can use Discord to discuss online. |
| 5 | Document | All meetings must be documented and pictured. |
| 7 | Slack | Store document resources and designed components |
| 8 | Discord | Discuss online, stream and share problems |

# REFERENCE

* [SDA]Proposal: <https://docs.google.com/document/d/1tQOYTQQ053oVq-IgPckF-CdjePMqiaZQ1QnrVysQJ8U/edit?usp=sharing>
* What is Scrum: <https://www.scrum.org/resources/what-is-scrum>