DUY TÂN UNIVERSITY
International School

CAPSTONE PROJECT 1

CMU-SE-450 / CMU-IS-450 / CMU-CS-450

PROJECT PROPOSAL

Version 1.2

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:	15 - Dec - 2020		
Lead Institution	International School, Duy Tan University		
Project Mentor	M.Sc 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 Proposal		
Author(s)	Team C1SE.06		
Role	[SDA] Proposal_v1.2		
Date	31 - Aug - 2020	File name	[SDA] Proposal_v1.2
URL	https://github.com/sdateamdtu2020/sda-documents		
Access	Project and CMU Program		

REVISION HISTORY

Version	Person(s)	Date	Description	Approval
Draft	Hoa, Vo	12 - Aug - 2020	Initiate proposal	x
1.0	All members	19 - Aug - 2020	Finish content of proposal	x
1.1	All members	31 - Aug - 2020	Update content & format	x
1.2	All members	16 - Nov - 2020	Update Tasks schedule	x

TABLE OF CONTENTS

1. INTRODUCTION

1.1. PURPOSE OF DOCUMENT

- Define the business needs and problems in detail.
- Provide solutions for business needs.
- Provide overview about resources, schedule, solution and budget for the project.

The proposal merely introduces the project to the student development teams, and provides the up-front information necessary for the team to develop a specification.

1.2. PROJECT GOAL

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.

1.3. BACKGROUND

Our environment is always changing. However, at the current rate of urbanization and industrialization, outside of the natural factors, the change of environment is mainly due to human factors. Emissions, population explosion, industrial solid waste, ... are the main causes leading to negative effects on the global environment. To address this at a holistic level, data analysis and aggregation are the first important tasks to be done.

However, analyzing and aggregating data from many different sources takes a lot of effort and money. To solve this problem, based on our knowledge of big data systems, we have built an intelligent data processing system that can be run on a website-platform with an intuitive and easy-to-use dashboard. This system is a prospective and useful tool for environmental experts and policy makers in Vietnam in particular, and worldwide in general. It will collect, analyze and synthesize data about all the factors that can affect the environment, thereby helping users to come up with quick and accurate solutions to solve problems that related to the environment.

2. PROBLEM DEFINITION

2.1. USER NEEDS

USER STORIES		
ID	**Actor**	**Epic**
1	As an environment expert	I want to drag a data cube from the cubes list and drop it onto the main content board, so that I can choose the property of data that I need and visualize them on suitable kinds of charts and maps.
2	As an environment expert	I want to drag some data cubes from the cubes list and drop them onto the main content board, so that I can choose the property of data and then connect these data cubes together for using data operation and visualize it on suitable kinds of charts and maps.
3	As an environment expert	I want to view the data source of any data cubes, so that I can verify the accuracy of the data.

2.2. NON-FUNCTIONAL REQUIREMENTS

Below are the non-functional requirements that are being offered for the system :

- **Portability and compatibility** : The system is operated on a web-based platform and has the ability to run on any web browser.
- **Security** : Users can use the system without the fear of revealing personal information.
- **Availability** : The system can run continuity 24/24 a day.
- **Usability** : The system has a friendly and flexible user-interface and great user-experience.
- **Reliability :** The system has accurate and transparent data, functions that do exactly its job.

2.3. FUNCTIONAL REQUIREMENTS

Below are the functional requirements that are being offered for the system, which are the backbone of the project :

- Drag data cubes from cubes list.
- Drop data cubes onto the main content board.
- Connect the data cubes that have the relation between them.
- Merge the cubes to create a new data cube based on connected data cubes.
- Visualize the data cube as a line chart, a pie chart, a column chart or a geographical map.

3. CURRENT STATUS OF ART

There are many research topics on data science applied to policy-making. Here are some researches that come from data scientist groups around the world.

1. Data science empowering the public (https://www.waze.com/): The Smart City dashboards in Rio de Janeiro, Brazil, were created to solve problems related to public transportation and traffic. For this an infrastructure, a dashboard, and a data portal with more than three thousand datasets and seven APIs for real-time data use (www.data.rio) were developed and used by the Center of Operations Rio (COR).
2. Interactive Dashboards (http://www.data.rio/): Using Visual Analytics for knowledge Transfer and Decision Support (Samar Al-Hajj, Ian Pike , Brian Fisher) -A visual analytics dashboard that reflects the needs and preferences of injury stakeholders. The types of visualizations were selected to efficiently illustrate trends and patterns in injury data.
3. Triangulum City Dashboard (https://moovitapp.com/): An Interactive Data Analytic Platform for Visualizing SmartCity Performance (Mina Farmanbar, Chunming Rong).

It can be seen that data science is increasingly weaving and useful in all areas of life, from urban management to public policy management and help in making decisions. But unfortunately, the world today is facing a vital problem, the environment. However, in today's market there are very few data analysis platforms to help policymakers make the right decisions about environmental protection.

This is the reason that makes us build a smart dashboard called Smart Dashboard Application - SDA, the first environment data analysis dashboard in the market now. With Smart Dashboard Application, the users can easily choose and integrate any data that are satisfied with their needs and visualize it in many kinds of charts and maps, so that they -

can consider the factors affecting the environment in the most comprehensive and intuitive way.

4. ENGINEERING APPROACH

4.1. CONTEXT DIAGRAM

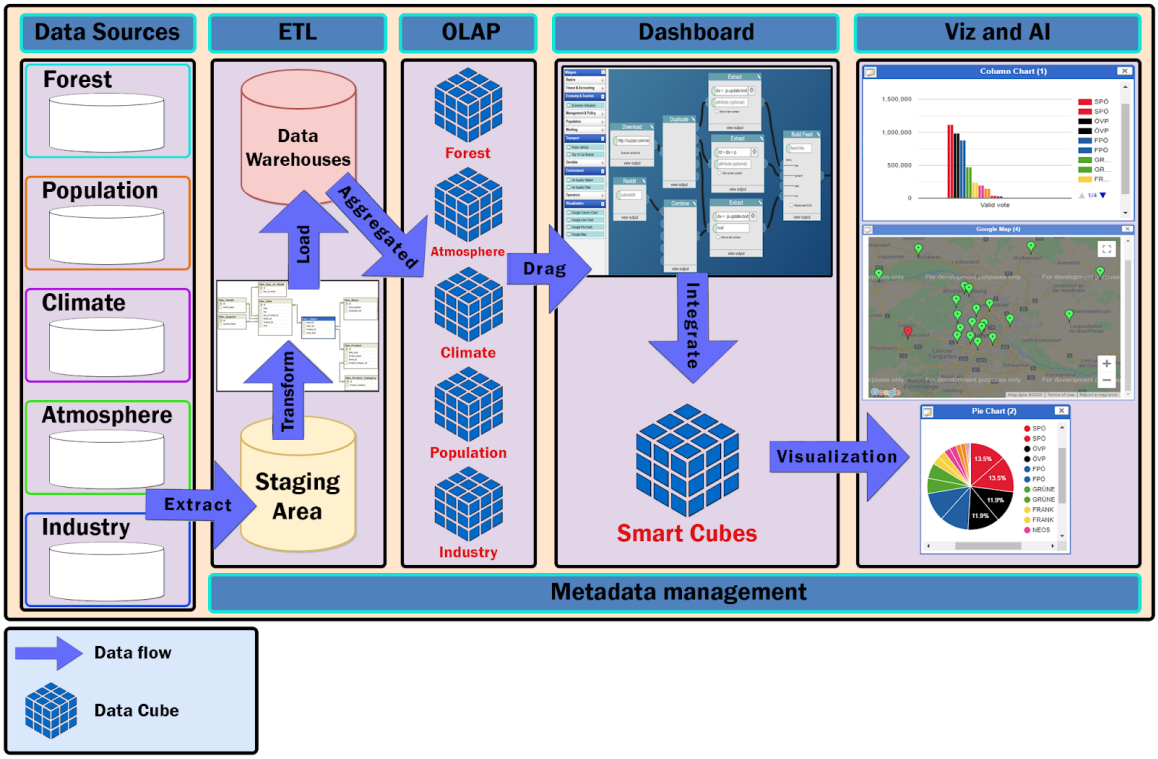


Figure 1 - 4.1: Context Diagram

4.2. PROCESS DETAILS

1. **Data Sources :**
   - Collect from environment open data platforms of the governments and NGO organizations.
   - Use web crawling techniques to crawl data from related environment websites.
   - Data Format : CSV, JSON, XML.
2. **ETL (Extract, Transform, Load)**

   2.1. Extract :

   In this step, data is extracted from the source system into the staging area. Transformations if any are done in the staging area so that performance of source

system is not degraded. Also, if corrupted data is copied directly from the source into the data warehouse database, rollback will be a challenge. Staging area gives an opportunity to validate extracted data before it moves into the Data warehouse.

2.2. Transform :

Data extracted from the source server is the raw data and not usable in its original form. Therefore it needs to be cleansed, mapped and transformed. In fact, this is the key step where the ETL process adds value and changes data such that insightful reports can be generated. In the transformation step, we filter , clean, split and integrate the data to match with the system requirements and data warehouse architecture.

2.3. Load :

Loading data into the target data warehouse database is the last step of the ETL process.

3. OLAP

In this step, we store the data of the data warehouse as OLAP cubes. And then, for better query performance, data binding and scalability, in addition for information transparency, we will convert OLAP cubes into RDF Data Cubes.

4. Dashboard

In this step, the user can drag any data cubes that appear as items on the sidebar and drop onto the main content board, then connect between them and use the operator such as statistics merge, geo merge to build a new data cube that matches the user requirement.

5. Viz & AI

This step will perform the data cube which was created by the user with the form they want. It can be a map, a column chart, a line chart or a pie chart.

4.3. TECHNICAL TO DEVELOP SYSTEM

Main Programming Language: Javascripts, Python.

Data Warehouses:

- Programming Language: Python.
- Database: PostgreSQL.
- Library: Psycopg2, CSV.

Data Cubes

- Programming Language: RDF-Graph, SPARQL.
- Tool for creating OLAP Cubes : SQL Server Analysis Services (SSAS).
- Tool for converting from OLAP Cubes to RDF Data Cube : OpenRefine.
- Network Accessing: RDF-REST API.

Server:

- Programming Language: Javascripts.
- Framework: ExpressJS (NodeJS).
- Libraries: Node-Postgres.
- Operating System: Windows, Linux, MacOS.
- Deployment Environment: Google Cloud with App Engine and SQL Services.
- Network Accessing: HTTP methods (POST, GET) via RESTful API.

Client:

- Programming language: HTML, CSS, Javascripts.
- Framework: React, Redux.
- Libraries: Material-UI, redux-Toolkit, react-dnd, beautiful-react-diagrams, highcharts
- Deployment Environment: Google Firebase Hosting
- Operating System: Windows, Linux, Mac OS.
- Web Browser: Chrome, Firefox, Microsoft Edge, Coccoc
- Network Accessing: World Wide Web (WWW), HTTP methods (POST, GET) via RESTful API with axios.

4.4. THE REASON BEHIND THE TECHNOLOGY STACK

Data Warehouses:

We choose PostgreSQL for the database. It tries to conform with the SQL standard where such conformance does not contradict traditional features or could lead to poor architectural decisions. Many of the features required by the SQL standard are supported, though sometimes with slightly differing syntax or function. Further moves towards conformance can be expected over time. As of the version 12 release in October 2019, PostgreSQL conforms to at least 160 of the 179 mandatory features for SQL:2016 Core conformance. As of this writing, no relational database meets full conformance with this standard.

Data cubes:

We choose to use the RDF model because it allows describing resources (either digital or taken from the real world), by specifying the values of their properties. Thus,an RDF information unit is a triple s p o, with s, p and o standing for the subject, property and object, respectively, in RDF terminology. The RDF language is increasingly being used in order to export, share, and collaboratively author data in many settings. For instance, it serves as a metadata language to describe cultural artifacts in large digital libraries, and to encode protein sequence data, as in the Uniprot data set. RDF is a natural target for representing heterogeneous facts contributed by millions of Wikipedia users, gathered within the DBpedia data source, as well as for the Linked Open Data effort, aiming at connecting and sharing collectively produced data and knowledge.

Server:

We chose ExpressJS to build the web server. It’s a prebuilt NodeJS framework that makes creating server side applications simple, fast, and flexible. NodeJS is powered by Google's V8 Engine which means it's powerful and can handle a large number of requests without lapsing in dependability. Also, this means that this is a highly scalable choice when you consider the Event Loop which manages all asynchronous operations allowing the program to continue to run as expected without stops.

Client:

We choose React to build the web client. React is developed by Facebook and right now, React and its ecosystems are the most popular stack for front end developers.

React introduces a new way to interact with the browser: Virtual DOM. We can now build web UI a lot faster than using the old one, traditional jQuery library. The UI has also become easier to develop and maintain. Beside that, React makes a huge impact on

performance, every change in the client side is now handled by the browser so that it will reduce server load. Together with Redux, React became an amazing tool: good performance, fast development and deployment speed, safe future. React has also got a huge and extremely active community that we can get help easily. And React-DnD is a set of React utilities to help you build complex drag and drop interfaces while keeping your components decoupled.

5. TASK AND DELIVERABLES

5.1. TASKS

WBS NUMBER	TASK TITLE
1	Preparation
2	Data Modeling
3	Physical Database Design
4	Setup Database
5	Finding Data
6	Data processing
7	Data Warehouse Designing
8	ETL Process Validating
9	Expanding database
10	Crawl data
11	RDF Data Cubes Designing
12	Setting up DW to RDF Process Tool & Environment
13	Implementing DW2RDF Process
14	Setting up RDF Data Cubes Storing & SPARQL Endpoint
15	Building a history data source
16	Validating RDF Data Cubes
17	Building SPARLQL_Rest API
18	Building UI
19	Testing
20	Integrate
21	Deploy
22	Release

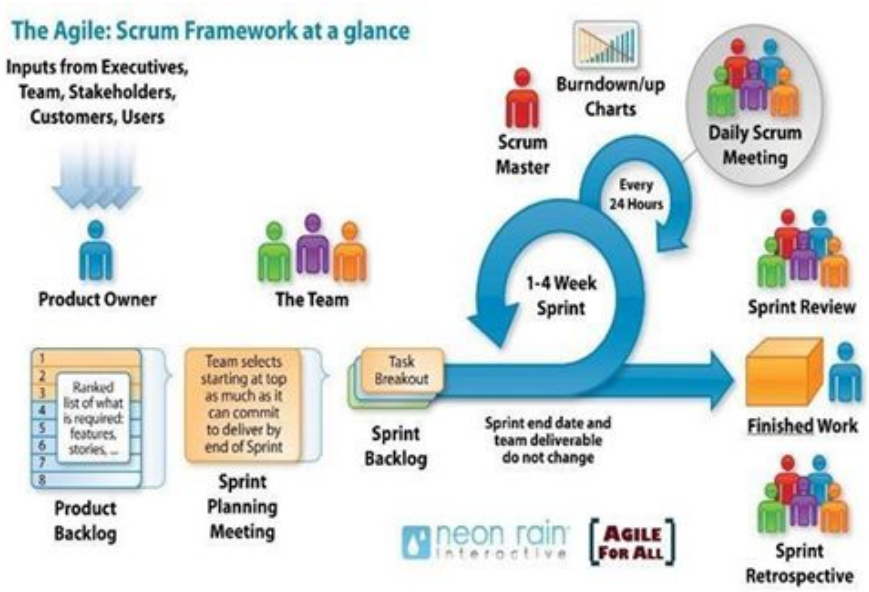
5.2. DELIVERABLES

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

6. PROJECT MANAGEMENT

6.1. ABOUT SCRUM

Scrum is an agile method, so it follows the principles of Agile Manifesto (see also Agile Manifesto). In addition, Scrum operates on three core values, also known as Scrip Scripps, including Scrutiny, Inspection and Adaptation.



Based on the empirical process control theory, Scrum uses iterative and incremental algorithms to optimize efficiency and control risk. Scrum is simple, easy to learn and has wide applicability. To be able to use Scrum, we need to understand and apply the elements that make up Scrum include the core values (also known as the "three legs", or the three pillars of Scrum), roles, Events, and Scrum-specific artifacts.

6.2. WHY SCRUM

- Our team has 4 people
- The project will be continuously horizontally scaled up.
- There is only a short amount of time to finish the project.

So based on those constraints, we decided to choose SCRUM as the project lifecycle.

6.3. COST PERSON/HOURS

Resource Name	Type	Max. Units	Std. Rate	Ovt. Rate
Hoa Vo	Work	100%	$2.00/hr	$3.00/hr
Tin Pham	Work	100%	$2.00/hr	$3.00/hr
Dong Ky	Work	100%	$2.00/hr	$3.00/hr
Kieu Tran	Work	100%	$2.00/hr	$3.00/hr

6.4. TOTAL COST ESTIMATE

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

6.5. DETAILED OF TASK ASSIGNMENT

WBS	Task	Start	End	Days
	Sprint 1	**Aug 12, 2020**	**Sep 10, 2020**	**30**
1	Data Modeling			
2	Physical Database Design			
3	Set up database			
4	Build UI			
5	Testing			
	Sprint 2	**Sep 11, 2020**	**Oct 10, 2020**	**30**
1	Finding Data			
2	Data processing			
3	Data Warehouse Designing			

4	ETL Process Validating			
5	Building UI			
6	Testing			
	Sprint 3	**Oct 11, 2020**	**Nov 9, 2020**	**30**
1	Expanding database			
2	Crawl data			
3	RDF Data Cubes Designing			
4	Setting up DW to RDF Process Tool & Environment			
5	Implementing DW2RDF Process			
6	Setting up RDF Data Cubes Storing & SPARQL Endpoint			
7	Building UI			
8	Testing			
	Sprint 4	**Nov 10, 2020**	**Dec 12, 2020**	**30**
1	Documents			
2	Building a history data source			
3	Validating RDF Data Cubes			
4	Building SPARQL-REST API			
5	Building UI			
6	Integrate			
7	Testing			
8	Deploy			
9	Release			

7. CONCLUSION

The project will be finished in 17 weeks and divided into 4 sprints. It promises to be a convenient and useful tool for not only environmental experts and policymakers but also those who love to protect the environment. The confusion about a large data system and reviewing it in a general way to come up with precise planning will no longer be a problem, even for those who don't have technology expertise. The project will contribute to help our society to accurately and effectively fight against environmental damage and global warming.

8. REFERENCE

- Software Development Standards for the Guidance and Control Software Project https://sw-eng.larc.nasa.gov/
- General Software Coding Standards and Guidelines https://www.nws.noaa.gov/oh/hrl/developers_docs/General_Software_Standards.pdf
- Scrum and best practices https://docs.microsoft.com/en-us/azure/devops/boards/sprints/best-practices-scrum?view=azure-devops
- The Scrum Guide https://www.scrum.org/resources/scrum-guide
- The ISO/IEC & IEEE/EIA Standard 12207, IEEE standards: IEEE-829 [3], IEEE-1008 [5], IEEE-1012
- Technologies Stack Document: https://docs.google.com/document/d/1DnWvniATIfAxsXl8dq2-REEw9L7tW5cYyz_W0GR6n9U/edit?usp=sharing
- React documentation : https://reactjs.org/docs/getting-started.html
- Redux documentation: https://redux.js.org/
- Draw tool: https://www.draw.io
- Data science empowering the public: Data-driven dashboards for transparent and accountable decision-making in smart cities: http://www.athanasiadis.info/assets/pdf/ems2018.pdf
- Smart cities survey: Technologies, application domains and challenges for the cities of the future: https://www.researchgate.net/publication/333684319_Smart_cities_survey_Technologies_application_domains_and_challenges_for_the_cities_of_the_future
- Interactive Dashboards: Using Visual Analytics for knowledge Transfer and Decision Support: https://www.researchgate.net/publication/299633100_Interactive_Dashboards_Using_Visual_Analytics_for_knowledge_Transfer_and_Decision_Support
- Dashboard technology based solution to decision making : https://www.researchgate.net/publication/277140671_DASHBOARD_TECHNOLOGY_BASED_SOLUTION_TO_DECISION_MAKING
- Triangulum City Dashboard: An Interactive Data Analytic Platform for Visualizing Smart City Performance: https://www.mdpi.com/2227-9717/8/2/250
- Data Science of the Natural Environment: A Research Roadmap: https://www.frontiersin.org/articles/10.3389/fenvs.2019.00121/full
- IEEE: https://www.ieee.org/
- QBOAirbase: European Air Quality Database as an RDF Cube: http://luisgalarraga.de/docs/qboairbase.pdf

- Enhanced Living Environments: https://link.springer.com/content/pdf/10.1007%2F978-3-030-10752-9.pdf
- An Approach To Publish a Data Warehouse Content as Linked Data: https://pdfs.semanticscholar.org/7913/6c3ad6f91aba1c9d186ede80fb1a1ec54704.pdf