Project Plan

MARKETPLACE

Ivanti

Date	**:**	**28/03/2022**
Version	**:**	**0.2**
State	**:**	
Author	**:**	**Group 2**

Version history

Version	Date	Author(s)	Changes	State
0.1	10/3/2022	All members	Initial version	Finish
0.2	28/03/2022	All members		Unfinished

Distribution

Version	Date	Receivers

Contents

1. Project assignment

Context

Ivanti Edge Intelligence is a platform for IT administrators retrieving facts from within their environment. With our platform the administrator can discover, gather insights, and take actions on happenings in their environment through software sensors. Characteristics about our platform are a simple user interface, fresh data as we retrieve our data directly from our endpoints, high performance, secure and scalable and we get better all the time with continuous delivery of new sensors, content, and visualizations. Our platform will be used all over the world, so localization and internationalization are key to success.
To make the customers of Ivanti able to access this platform and all other products of Ivanti a marketplace is required. More about this is the goal of the project.

Goal of the project

Our project goal is to extend the capabilities (sensors) within Edge Intelligence with a new platform that provides customers (new) content through a marketplace. Our marketplace will provide customers with added content delivered by Ivanti, partners or the community, they eventually can install into their environment. We believe a great customer experience will drive the use of the marketplace and will deliver more external content creators towards our platform.

Scope and preconditions

Inside scope:	**Outside scope:**
1 Two types of users (Creators & Customers)	1 Actual downloading of packages
2 Internationalization & Localization	2 Upload and update the actual packages

Strategy

We will approach this project using agile and scrum. We are going to do 5 sprints, each 3 weeks long, with a meeting with the stakeholder at the end of every sprint as well as a developer team meeting to reflect on the sprint and plan the approach for the next one.

Research questions and methodology

For research we will be using document analysis and an interview to grasp more understanding of this project. For the actual implementation of the application's back and front end we will use Community research.

For localization, as a user you will be able to change languages. Changing languages sounds easy at first but we aim to add another language easily. Our goal is not to translate the page manually for every word but to do it automatically to some extent. More information can be found in research documents for translations.

End products

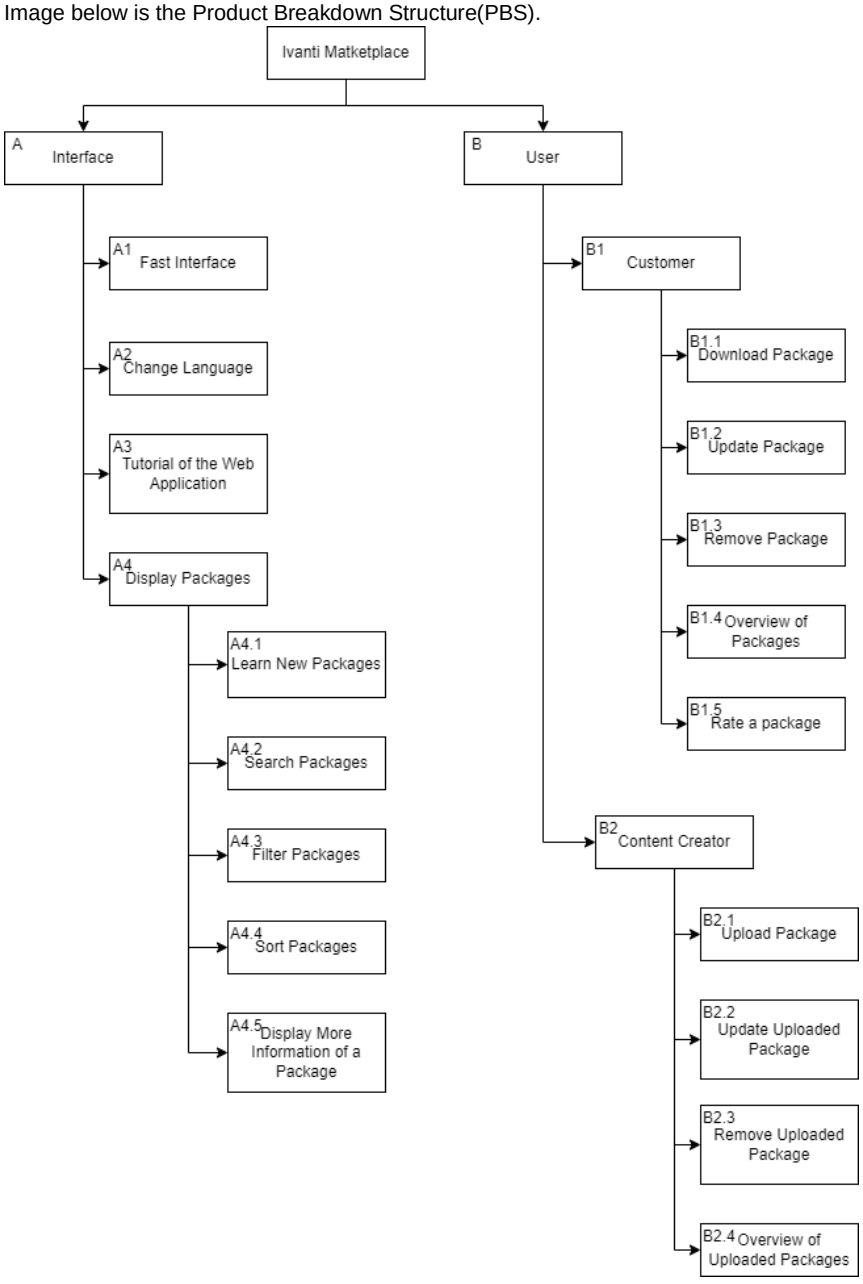
The product contains a web application that allows the user to access a kind of marketplace.
Below is the table of user stories that will be implemented in the web application.

No.	User Story	Estimation	Sprint	Priority
1	As a customer when navigating to the marketplace I would love to learn more about how to use the marketplace	40	E	20

	Acceptance criteria: • Show an interactive tutorial that explains key features.			
2	As a customer when navigating to the marketplace I would love to learn about available content **Acceptance criteria:** • Home page with introduction to the app is displayed • Information regarding policies and content is displayed • Information about distinct roles of users is displayed	80	B/C	80
3	As a customer when finding the right content, I would love to read more about this content **Acceptance Criteria:** • Show package description and other information such as category and content creator name are shown. (All package information) • Show install/update/delete button. • Show the rating and reviews. • Show a button for leaving a review.	70	C	60
4	As a customer when navigating to the marketplace I would love to see the content sorted a – z **Acceptance criteria:** • Availability to sort content on different properties • Content is displayed according to sorting criteria	30	D/E	20
5	As a customer when looking for something specific I would love to have the ability to search for content **Acceptance criteria** • Let the user enter text in a search bar. • Return a list of packages that contain the user search query in the package name.	30	D/E	20
6	As a customer when navigating it would be helpful when content is grouped so that I can find content faster **Acceptance criteria:** • Let the user choose a filter. • Return a list of packages based on the selected filter.	30	D/E	20
7	As a customer when navigating I would expect a fast user interface when thousands of content packages are available **Acceptance criteria:** • Show only a limited number of packages on the page. • Let the user choose to show the next number of packages (page navigation).	50	E	70
8	As a customer I would love to read content in my local language so that I can enjoy the platform more **Acceptance criteria:** • Language is rendered depending on the location of user • Availability to change language if necessary • Multiple languages support	50	B-E	60
9	As a customer I would love to add packages to my environment so that I can use the capabilities in my environment **Acceptance criteria:** • A button to download the package. • A brief description and capabilities of the package.	60	B	90

10	As a customer I would love to have an overview of all packages that I added to my environment **Acceptance Criteria:** • A table of packages that have been added to the library is shown • Number, package name, category, and update check are displayed • There are search and sorting features for this table • In the update check, if there is no update, "Open" button is displayed and if there is an update, "Update" button is displayed and you if you click on the update, it will automatically update your current app and after the update finish, it will change to "Open" button	60	C	60
11	As a customer I would like to leave a review and rate a package so that I can let the package creator know if the package is good. **Acceptance criteria:** • Show a 1 to 5 starts input field for the rating. • Show an input field for the review. • Show a submit button for submitting the rating. The user can only leave one review per package.	50	E	10
12	As a creator of content, I need an area to upload added content so that this becomes available in the marketplace **Acceptance criteria** • Input fields for, title, description, category. • Upload field for the file. • Show a submit button to submit the package.	50	B	100
13	As a creator of content, I need the ability to update my content so that I can supply the latest updates **Acceptance criteria:** • Possibility to update content • Creators can update only the content they uploaded • Creators are not able to update other creators' content	40	C	80
14	As a creator of content, I need an overview of my content so that I can easily manage all the content that is from myself **Acceptance criteria:** • A table which displays all the content the creator has uploaded. • I can click every package to change information of the package.	60	C	60
15	As a user, I should be able to sign up so that I access customer and/or creator of content features. **Acceptance criteria:** • Text inputs are shown • Input fields includes, email, password, and repeat password • A sign-up button can be click to submit the registration	40	D	50
16	As a user, I should be able to sign in so that I access customer and/or creator of content features. **Acceptance criteria:** • Text inputs are shown • Input fields includes, email and password • A sign in button can be click to submit the sign in form	30	D	50

Image below is the Product Breakdown Structure(PBS).



PBS above is divided into 2 main products, the interface (A) and the users (B).

Interface consist of 4 sub-products. This web application needs a fast interface (A1), users can change the language (A2) to which they want and available in the web application, a tutorial of the web application (A3) for the key features that are needed to be explained, and on how to display packages (A4). There are several ways on how packages can be displayed. The first is the default option where the web application recommend packages so that users can learn new packages (A4.1). User can also edit on how they want to view packages by searching the packages (A4.2), filter the packages (A4.3), and sort the packages (A4.4). User can look more information of the chosen package (A4.5) where it displayed full information of the package.

User consist of 2 sub-products, customer (B1) and content creator (B2). Customer can download a package (B1.1), update the downloaded package (B1.2) if the content creator decided to update the package, remove a package (B1.3), see overview of their downloaded packages (B1.4), and rate a package (B1.5) which the customer has downloaded. Content creator has 4 sub-sub-products, it includes, upload new package (B2.1), update the uploaded package (B2.2), remove uploaded package (B2.3), and they can see the overview of their uploaded packages (B2.4).

2. Project organisation

Stakeholders and team members

Name	Abbreviation	Role and functions	Availability
Dennis Smits	*Mr.*	*Product owner*	*Undefined time slots, reachable by email and teams.*
Airell Rasendriya Bachtiar airell.bachtiar@student.fontys.nl	*Mr.*	*Developer*	*Every day from 9:00 to 18:00*
Borek Bandell b.bandell@student.fontys.nl	*Mr.*	*Developer*	*Every day from 9:00 to 16:00*
Nick can der Meer nick.vandermeer@student.fontys.nl	*Mr.*	*Developer*	*Every day from 09:00 to 16:00, always reachable by teams or discord*
Nikola Stankov n.stankov01@gmail.com	*Mr.*	*Developer*	*Every day from 9:00 to 16:00*
Roel Habers 460913@student.fontys.nl	*Mr.*	*Developer*	*Always reachable will respond asap.* *Reachable by mail, teams, or discord*
Viktor Skachkov v.skachkov@student.fontys,nl	*Mr.*	*Developer*	*Every day from 9:00 to 20:00 o' clock.*

Communication

We will meet each week two or three times. If possible, the meetings will be in the university, so that we can discuss our progress as well as innovative ideas face to face. Sometimes, we will have meetings online if we need to discuss something urgently (especially during the weekends).

We will contact the company supervisor, Dennis Smits, via email or Teams if we have any questions and we will meet him once per sprint at the end, so that we can show our progress. We will have meetings with the teachers each week, so that they can see our progress and give feedback.

3. Activities and time plan

Phases of the project

We will work with sprints. A sprint consists of:

- Product Backlog
- Sprint planning meeting
- Sprint backlog
- The actual development
- Standups
- Sprint review

Before each sprint starts, we need to make a product backlog and sprint backlog. At the start of each sprint, we have a sprint planning meeting. During this meeting we discuss the work that needs to be done and we try to divide the work.
After this the development time starts. We have standup meetings around one or two times each week to discuss with each other what work we have done and discuss the problems we might be facing.
When the development is done, we have a sprint review with the product owner to show the work we have done this sprint and reflect on the process. Then we have a retrospective meeting. In this sprint the team itself discusses what went right and what could be improved in the next sprint.
In Sprint A, we will focus on the documentation. This includes project plan, software architecture design, etc. Ideally, we will start implementation in sprint B. Documents will be updated throughout the project. Changes can be made in the middle of the project. Testing should be done through out the project so that it will save time at the end of the project.

Time plan and milestones

Phasing	Start date	Finish date
1 Sprint A	07.03.22	25.03.22
2 Sprint B	28.03.22	14.04.22
3 Sprint C	18.04.22	13.05.22
4 Sprint D	16.05.22	03.06.22
5 Sprint E	06.06.22	24.06.22

4. Testing strategy and configuration management

Testing strategy

We will use two types of testing, unit testing and user acceptance testing.

Unit testing: 75% of our code will be covered by unit tests.
Our back end is written in Java. For testing this Java code, we will use JUnit. JUnit is a unit testing framework for Java.
For testing the front end, we will use user acceptance testing.

Test environment and required resources

We will make use of the CI/CD environment provided by Gitlab. We will have runners on local devices. When someone makes a commit to the repository one of these runners will execute the unit tests on this latest commit. If the tests fail the pipeline fails and the users will be notified. They will know the latest commit broke the code and can fix this as soon as possible. This way bugs in the code cannot sneak into our codebase without us noticing.
We will use SonarQube to check our code quality and with the help from JaCoCo, it will help with the test coverage from the unit test.

Commented [AA1]: *Use SonarQube*

Configuration management

We will be working with Git. The project will be hosted on the Fontys hosted Gitlab. We will divide our branch based on features and a development branch which is a branch that will be combined from features branches before pushing to the master branch. Master branch will act as a final release of our application.

5. Finances and risk

Project budget

We do not have a budget for this project. We will use resources provided by Fontys.

Risk and mitigation

Risk	Prevention activities	Mitigation activities
1 One or more of us decide to drop out of the project.	We will work as a team and help each other to make sure that no one would feel out of place in the team.	We will all redistribute the work to make up for the decrease in people in our team.
2 We do not have enough time to finish our work before the end of the sprint.	We start working earlier, so that we have enough time to finish everything even if something unexpected occurs.	Each of us starts working as hard as possible without stopping, so that we can finish on time.
3 We do not have enough knowledge to do our work.	We will find courses or other sources to learn the things we need.	We will improvise with the code and do the features with code we know.

4	One or more of us gets ill and we cannot reach them.	We will try to lead a healthy lifestyle, so that none of us gets ill.	We will stay connected via email, Discord, Teams, or other means, so that the people who are sick can still take part in the group effort.
5	Product owner is unavailable	-	Postpone activities that need product owner guidance