

Project plan for IAS

BluDevil digital marketplace



Members: Alexandru, Emilian E.M.

Tutors: Geurts, Jaap J. ; Guayrin, Brice B.P.B.J.P.

Version: 1.1

Table of contents:

Table of contents:	2
1.Project description	4
2.User Stories	4
3.Product backlog	10
4.Sprint planning	12
Sprint 1	12
Sprint 2	13
Sprint 3	13
Sprint 4	14
Sprint 5	14
Sprint 6	15
5.API documentation	16
5.1 Data description	16
5.2 RESTful API design	19
6.Version history	22
6.1 Document version history	22
6.1.1 Project plan	22
6.1.2 Design document	23
6.2 API version history	24

1. Project description

The project is a digital marketplace named BluDevil, a web application for selling and managing digital products. Each product(ex. Software product, video game...) will contain a cd key.

The application contains of:

- A product managing system where an administrator can create, view a product and modify its data and availability.(available, unavailable)
- A user managing system where an administrator can create, view, remove and modify a user's data and state(active, inactive)
- A customer chat support system.
- A shopping cart where a user can add and remove products to later proceed to checkout.

2. User Stories

The MoSCoW prioritisation method is applied for the user stories priority.

M(MUST), S(SHOULD), C(COULD), W(WOULD)

US-Nr	User story	Acceptance criteria	Time estimate	Priority
US-01	<ul style="list-style-type: none">• As a visitor• I can create an account• So that I can become a member	<ul style="list-style-type: none">• I can enter my first name, last name, email and password to create an account.• When entering my password I must enter it again to confirm it.• I received an email confirmation of my registration.• The system notifies the user if the email is already in use.• The system notifies the user if the validation failed• A message is displayed if the registration failed.	2 days	M
US-02	<ul style="list-style-type: none">• As a account holder• I can login into the application• So that I can access more services	<ul style="list-style-type: none">• I can enter my credentials(email and password) to login• I receive an error message if try to log in with false credentials• I receive an error message if my email or password are incorrect.	2 days	M

		<ul style="list-style-type: none"> I am successfully logged in and redirected as a user to the index page and as an administrator to the administrator panel. 		
US-03	<ul style="list-style-type: none"> As a user I can see all the products in the marketplace So that I know the available products 	<ul style="list-style-type: none"> The products are displayed by pagination A product is displayed as a card with the name, image and platform. I can navigate to the first,previous,next and last page of the products. The system displays a error if there are no products 	4 days	M
US-04	<ul style="list-style-type: none"> As a user I can search products by the product name So that I can find a specific product 	<ul style="list-style-type: none"> I can use the search bar to enter a product name The products that match the user input are displayed by pagination. The products are not displayed if no match was found and a message is provided. I can navigate to the first,previous,next and last page of the searched product. 	1-2 days	M
US-05	<ul style="list-style-type: none"> As a user I can filter by price and/or platform all the products So that I can find desired products easier 	<ul style="list-style-type: none"> The products matching the active filter or filters(price and platform) are displayed by pagination. The products are not displayed if no match was found and the system displays an error message. I can navigate to the first,previous,next and last page of the filtered products. 	1-2 days	S
US-06	<ul style="list-style-type: none"> As a user I can inspect a product So that i can view all its details 	<ul style="list-style-type: none"> After clicking a product card I am redirected to another page where all the data(name, image, availability status, description and platform) from the selected product is displayed. Depending on the type of product other information might be shown(example videogame type will also display the release date and genres) 	2 days	M
US-07	<ul style="list-style-type: none"> As a member I can use the cart So that i can proceed to the checkout 	<ul style="list-style-type: none"> I can see my products in the cart. I can remove the selected product in the cart or decrease its quantity. 	3-4 days	M

		<ul style="list-style-type: none"> I can increase the selected product from the cart by pressing a button. If no product is selected in the cart no operation can be performed. When inspecting a product I can add a product to the shopping cart. 		
US-08	<ul style="list-style-type: none"> As a member I can see my account settings So that I can edit my information 	<ul style="list-style-type: none"> The user's first name, last name, email, password is displayed in fields. The user input is validated. I can see if the entered data is wrong. The system notifies the user if the email is already in use. I am notified if the changes fail. I can change my first name, last name, email and password. When changing my password I have to enter it two times 	2 days	M
US-09	<ul style="list-style-type: none"> As a member I can add and remove products in my wishlist So that I can look at them later 	<ul style="list-style-type: none"> When inspecting a product I can add a product to my wishlist. When inspecting my wishlist I can remove a selected product. All the wished products are displayed by pagination I can click a product to view its data(name, image, availability status, description and platform) and the system redirects me to selected product page 	2 days	W
US-10	<ul style="list-style-type: none"> As a member I can use the support chat So that I can say my problems 	<ul style="list-style-type: none"> I can open the support chat. System creates a chat room. I can write and receive a message. I can see if an admin is available to chat. I can get notified when an admin left the chat 	3-4 days	S
US-11	<ul style="list-style-type: none"> As a member I can see my orders So i can inspect one 	<ul style="list-style-type: none"> The orders are displayed sorted by creation time. I can select an order to inspect it. The order date, products names, total price and products cd-keys are displayed when inspecting an order. I can navigate to the first,previous,next and last page of the orders. 	3 days	M

US-12	<ul style="list-style-type: none"> As a administrator I can access the dashboard So I that I can see various statistics 	<ul style="list-style-type: none"> The total number of users is displayed. The average users registered per week is calculated and displayed. The total number of products is displayed. The product turnover is calculated and displayed for a selected product. The five most popular products, sold during the week of a given date, are displayed. 	2-3 days	S
US-13	<ul style="list-style-type: none"> As a administrator I can view all products So that I can manage them 	<ul style="list-style-type: none"> The products are displayed in a pagination. I can navigate to the first,previous,next and last page of the products. I can add a new product. The system redirects the user to the product creation page where all the product data needs to be filled. I can enter the data(name, image, availability status, description, platform and availability of a product) when creating a product. Depending on the selected type of product I can add other information like release date and genres for a videogame The user input is validated. I get notified if the creation was not successful. I get redirected to the products manage page if the changes were successful. 	2-3 days	M
US-14	<ul style="list-style-type: none"> As a administrator I can view all users So that I can manage them 	<ul style="list-style-type: none"> The users are displayed by pagination. I can navigate to the first,previous,next and last page of the users. I can create a new user. The system redirects the user to the user creation page if the create button is pressed. I can enter the data(first name, last name, email, password and role). I have to enter the password two times. I get notified if the data i enter is not valid or if the creation failed I get redirected to the user manage page on successful creation 	2-3 days	M

US-15	<ul style="list-style-type: none"> As a administrator I can see the support chats So that I can respond to them 	<ul style="list-style-type: none"> The user chats are displayed, sorted by creation date, by pagination. I can select a chat-room to view it and join to respond to the user. I can leave from a chat room. 	3-4 days	M
US-16	<ul style="list-style-type: none"> As a administrator I can view a products cdkeys So I can manage them 	<ul style="list-style-type: none"> After selecting a product from the product manage page I can view the cdKeys of the product by pagination. The system redirects the user to the cdKey manage page of the selected product where a cdkey can be created or updated. I can enter the cdKey sequence when editing and creating a key. The system validates the user input. I get notified if the action failed or if the input is not valid. After a successful action i am redirected to the selected product cdKey manage page. 	2-3 days	M
US-17	<ul style="list-style-type: none"> As a administrator I can view the products platforms So I can manage them 	<ul style="list-style-type: none"> I can view all the platforms by pagination I can edit or create a new product platform The system redirects the user to the product platform creation or edit page. I can enter the product platforms data (name) The data is validated by the system. I get notified if the action was not successful by a message. I get redirected to the product platforms manage page upon a successful action. 	2-3 days	M
US-18	<ul style="list-style-type: none"> As a administrator I can select a product So I change it 	<ul style="list-style-type: none"> I can select a product from the list to edit or delete. The system redirects the user to a product edit page where all the product data is displayed in fields. I can change the data(name, image, availability status, description, platform and availability of a product). Depending on the type of product I can change other information like release date and genres for a videogame 	2-3 days	M

		<ul style="list-style-type: none">• The user input is validated.• I get notified if the changes were not successful.• I get redirected to the products manage page if the changes were successful.		
US-19	<ul style="list-style-type: none">• As a administrator,• I can select a user,• So I can edit their data	<ul style="list-style-type: none">• I can select a user to change his data or remove him.• The system redirects the user to the user editing page where all the user data is displayed in fields.• I can change the data(first name, last name, email, password, role and state) of the user in the user edit page.• I get notified if the data i enter is not valid or if the changes failed	2-3 days	M

3.Product backlog

Legend items:

 User story

Legend items status:

 Done  WIP(Work in progress)  TBS(To be started)

US-Nr	Story	Priority	Sprint	Status	Estimated time
03	As a user, I can see all the products in the marketplace, so that I know the available products	M	2	Done	4 days
06	As a user , I can inspect a product , so that i can view all its details	M	2	Done	2 days
04	As a user, I can search products by the product name, so that I can find a specific product	M	2	Done	1-2 days
05	As a user, I can filter by price and/or platform all the products, so that I can find desired products easier	S	2	Done	1-2 days
13	As a administrator, I can view all products, so that I can manage them	M	2	Done	2-3 days
18	As a administrator, I can select a product, so I change it	M	2	Done	2-3 days
16	As a administrator, I can view the products platforms, so I can manage them	M	2	WIP	2-3 days
17	As a administrator, I can view a products cdkeys, so I can edit them	M	3	TBS	2-3 days
07	As a member, I can use the cart, so that i can proceed to the checkout	M	3	TBS	3-4 days
11	As a member, I can see my orders, so i can inspect one	M	3	TBS	3 days

14	As a administrator, I can view all users, so that I can manage them	M	3	TBS	2-3 days
19	As a administrator, I can select a user, so I can edit their data	M	3	TBS	2-3 days
12	As a administrator, I can access the dashboard, so I that I can see various statistics	S	4	TBS	2-3 days
01	As a visitor, I can create an account, so that I can become a member	M	4	TBS	2 days
02	As a account holder, I can login into the application, so that I can access more services	M	4	TBS	2 days
08	As a member, I can see my account settings, so that I can edit my information	M	4	TBS	2 days
10	As a member, I can use the support chat, so that I can say my problems	M	5	TBS	3-4 days
15	As a administrator, I can see the support chats, so that I can respond to them	M	5	TBS	3-4 days
09	As a member, I can add and remove products in my wishlist, so that I can look at them later	W	5	TBS	2 days

Total sprint 2: 19 days

Total sprint 3: 16 days

Total sprint 4: 8 days(+possible additional due to previous week requirements)

Total sprint 5: 10 days

4.Sprint planning

The project lifecycle is 17 weeks and it is divided into six sprints. Starting from the 1st till the 5th sprint, each sprint will have a duration of 3 weeks. The last sprint duration is 2 weeks. In this section the activities and deliverables for each sprint are enumerated.

Sprint 1

Activities:

- Working on the project planning document
- First setup of Restful API
- CI environment initialization
- Creating wireframe/sketches for the user interface
- Start creating ERD diagram

Derivables:

- Project plan document
- Restful API for web application
- CI environment

Sprint 2

Activities:

- Create entity relation diagram
- Create high level architecture respecting SOLID
- Create UML class diagrams
- Start creating frontend user interface based on wireframe/sketches
- Start developing the backend of the application and tackle the user stories based on their priority
- Finish 1st prototype

Derivables:

- 1st version of the design document
- 1st prototype

Estimated completed user stories: US-03, US-04, US-05, US-06,US-13, US-16

Sprint 3

Activities:

- Continue working on design document
- Continue working on the functionalities
- Finish 2nd prototype

Derivables:

- 2nd version of the design document
- 2nd prototype

Estimated completed user stories: US-14, US-07, US-08, US-11, US-13,US-19

Sprint 4

Activities:

- Work on design document
- Polish working functionalities
- Set up backend in multiple layers using Interface segregation and dependency inversion
- Finish 1st release of the application
- Create a movie per finished user story
- Work on sprint backlogs and burn-down charts
- Work on quality assurance metrics
- Present application, receive feedback from two users and create the UX feedback report

Derivables:

- 3rd version of the design document
- 1st release version
- Sonarqube screenshot/report pre-post
- UX feedback report

Estimated completed user stories: US-08, US-01, US-02, US-12

Sprint 5

Activities:

- Finish design document
- Apply user feedback from UX report
- Finish 2nd release of the application
- Work on backlogs and burn-down charts

Derivables:

- Final design document
- 2nd release version
- Updated backlogs and burn-down charts

Estimated completed user stories: US-10, US-15, US-09

Sprint 6

Activities:

- Work on final report
- Work on final release with docker integration
- Update backlogs and burn-down chart
- Hand in all the work

Derivables:

- Reading guide
- All feedback form individual track and group project in PDF
- Updated backlogs and burn-down chart
- Final release

5.API documentation

5.1 RESTful API design

(The greyed out URLs are temporary, possible changes to the current urls by making use of URL queries)

Products User Orders Chat Other

Operation	URL	Description
GET	api/products/{id}	Get the product by id.
POST	api/products	Create a new product.
PUT	api/products/{id}	Update a product.
DELETE	api/products/{id}	Delete a product by its id.
GET	api/products?page={pageNr}&size={size}	Get all products with pagination.
GET	api/products?page={pageNr}&size={size}&price={price}	Get all products with pagination by the given price.
GET	api/products/platform/{platformName}?page={pageNr}&size={size}	Get all products with pagination by given platform name.
GET	api/products/platform/{name}?page={pageNr}&size={size}&price={price}	Get all products with pagination by given price and platform name.
GET	api/products/search/{productName}?page={pageNr}&size={size}	Get all products with pagination by the given name.
GET	api/products/statistics	Get the general statistics of the products. Returns, the total number of all products, the total number of

		the products by platform.
GET	api/products/statistics/{id}	Get the product turnover of the product with the given id.
GET	api/products/statistics/top/{yyyy-MM-dd}	Get the top 5 most sold products during the week of the given date.
GET	api/productPlatforms?retrieval={mode}	Get all product platforms.
GET	api/productPlatforms?page={pageNr}&size={size}	Get all platforms by pagination.
GET	api/productPlatforms/{id}	Get a product platform by id.
POST	api/productPlaforms	Create a new product platform.
PUT	api/productPlarforms/{id}	Update a product platform.
GET	api/cdKeys/product/{id}?page={pageNr}&size={size}&used={isUsed}	Get all available or unavailable cdKeys with pagination by a product id.
GET	api/cdKeys/{id}	Get a cdKey by its id.
POST	api/cdKeys	Create a new CdKey for a product.
PUT	api/cdKeys/{id}	Update a CdKey for a product.
DELETE	api/cdKeys/{id}	Delete a CdKey by its id.
GET	api/genres?page={pageNr}&size={size}	Get all genres by pagination.
GET	api/genres?retrieval={mode}	Get all genres.
GET	api/genres/{id}	Get a genre by its id.
PUT	api/genres/{id}	Update a genre by its id.

POST	api/genres	Create a new genre.
POST	api/images/upload	Upload a jpeg,png,gif image.
GET	api/images/getImage/{imageName: .+}	Get an image by name.

6.Version history

6.1 Document version history

6.1.1 Project plan

Version/Date	Description
V1.0 / 04-03-2021	<ul style="list-style-type: none">• Setup project layout• Wrote project description• Wrote user stories• Wrote product backlog• Created temporary API data description• Wrote API design for the “products” section and the images for “other” section• Wrote version history for the project plan(v1.0) and the API (v1.0.0)
V1.1 / 07-03-2021	<ul style="list-style-type: none">• Removed data description section• Modified api endpoints
V1.2 /16/04/2021	<ul style="list-style-type: none">• Updated product backlog• Update versioning sections

6.1.2 Design document

Version/Date	Description
V1.0 / 04-03-2021	<ul style="list-style-type: none">• Created ERD based on temporary data description from Project Plan V1.0 (currently contains products, users and orders data)
V1.1 / 10-03-2021	<ul style="list-style-type: none">• Updated the ERD to contain a legend for cardinality component (relationships) V1.1• Created UML diagram V1.1• Created wireframes for product listing and product details• Wrote design decisions
V1.2 / 16/04/2021	<ul style="list-style-type: none">• Modified diagrams to match new implementations• Splitted diagrams into sections• Added a descriptive text for all the diagrams• Added data persistence section• Added sonarqube analysis section• Updated the Front-end frameworks section from the design decision

6.2 API version history

Version/Date	Description
V1.0.0 / 04-03-2021	<ul style="list-style-type: none">• Created the directories structure• Created the product entity• Added the VideoGame and SoftwareProduct as child entities that extend the product entity• Created the ProductRepository that extends the JpaRepository• Created the product service• Created the product controller• Implemented the product CRUD• Implemented the product pagination with filters• Created the ProductPlatform entity• Created the ProductPlatform Repository that extends the PagingAndSortingRepository• Created the product platform service• Created the product platform controller• Created the Genre entity• Created the GenreRepository that extends the PagingAndSortingRepository• Created genre service• Create genre controller• Created the MessageDTO and Pagination responses• Created two unit tests with JUnit4, one for the ProductController CRUD and the other for the ProductPlatformController CRUD• Created the image service and image controller• Implemented uploading an image (png, jpeg, gif)• Implemented getting an image by the file name
V1.0.1/ 04-03-2021	<ul style="list-style-type: none">• Refactored code based on sonar pulse analysis• Reorganised test directory structure and added the first unit test• Tackled the high number of codesmells• Updated the product section by implementing filtering and modifying the product retrieval based on the product state• Implemented CRUD for genres and product platforms• Created exception adviser