



Aplicações na Web
2021/2022
Mestrado em Engenharia Informática

Project Blueprint

Rui Roque nº57588
Tomás Dias nº57582

**Front-end (Presentation Logic)
&
Back-end (Business Logic)
Handlers**

Microservices		Request			Response object
		HTTP Call	URI	Request object	
Front-end	Create Account	POST	/user	JSON Object: User	JSON string: code string: message Result description: Create user
	Login	POST	/user/login	JSON string: username string: password	JSON string: code string: message Result description: Logs user into the system
	Logout	GET	/user/logout	Empty	JSON string: code string: message Result description: Logs out current logged in user session
	User Profile	GET	/user/{user_id}	Empty	JSON Object: User Result description: Get user

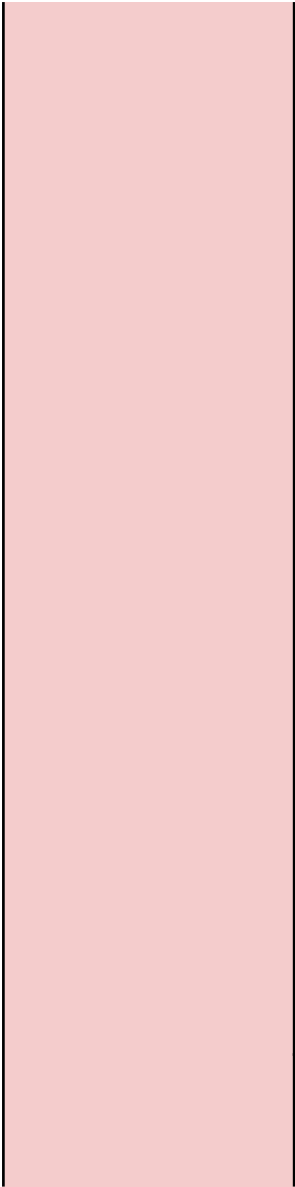
	Update User Profile	PUT	/user/{user_id}	JSON Object: User	JSON string: code string: message Result description: Update user
	Delete User Profile	DELETE	/user/{user_id}	Empty	JSON string: code string: message Result description: Delete user
	Add Building	POST	/building	JSON Object: Building	JSON string: code string: message Result description: Add a new Building to the App
	Buildings	GET	/building/{user_id}	Empty	JSON Array of Objects: Building Result description: Get List of Buildings By User

	Building	GET	/building/{building_id}	Empty	JSON Object: Building Result description: Get Building
	Update Building	PUT	/building/{building_id}	JSON Object: Building	JSON string: code string: message Result description: Update Building
	Delete Building	DELETE	/building/{building_id}	Empty	JSON string: code string: message Result description: Delete Building
	Add Work	POST	/work	JSON Object: Work	JSON string: code string: message Result description: Add a new work to a building

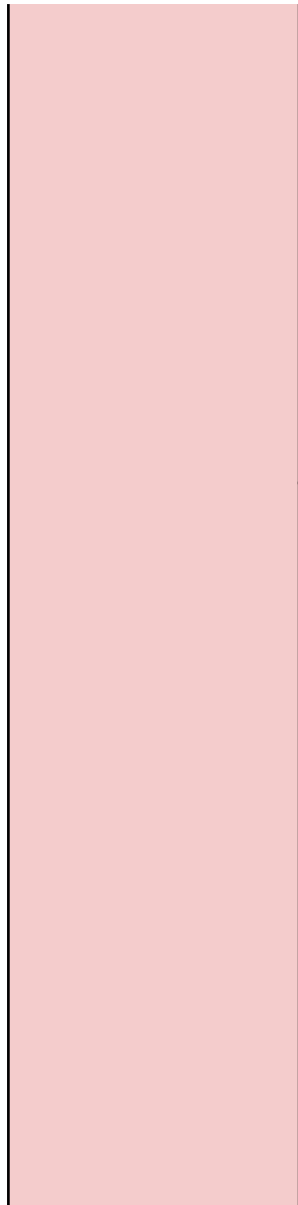
	Works List By Building	GET	/work/{building_id}	Empty	JSON Array of Objects: Work Result description: Get list of works by building
	Works List By User	GET	/work/{user_id}	Empty	JSON Array of Objects: Work Result description: Get list of works by user
	Work	GET	/work/{work_id}	Empty	JSON Object: Work Result description: Get a building work
	Update Work Status	PUT	/work/{work_id}/updateStatus	JSON: string: status	JSON string: code string: message Result description: Update a building's work status

	Update Work	PUT	/work/{work_id}	JSON Object: Work	JSON string: code string: message Result description: Update a building work
	Delete Work	DELETE	/work/{work_id}	Empty	JSON string: code string: message Result description: Delete a building work
	Add Feedback	POST	/feedback	JSON Object: Feedback	JSON string: code string: message Result description: Create feedback from a work
	Feedbacks List By Work	GET	/feedback/{work_id}	Empty	JSON Array of Objects: Feedback Result description: Get work's list of feedbacks

	Feedback	GET	/feedback/{feedback_id}	Empty	JSON Object: Feedback Result description: Get a work feedback
	Update Feedback	PUT	/feedback/{feedback_id}	JSON Object: Feedback	JSON string: code string: message Result description: Update a work feedback
	Delete Feedback	DELETE	/feedback/{feedback_id}	Empty	JSON string: code string: message Result description: Delete a work feedback
Back-end	Manage users	POST	/user	JSON Object: User	Create user
		GET	/user/{id}	Empty	Get user



			JSON Object: User	
	PUT	/user/{id}		Update user
	DELETE	/user/{id}	Empty	Delete user
	POST	/user/login	JSON string: username string: password	Logs user into the system
	GET	/user/logout	Empty	Logs out current logged in user session
Manage Buildings	POST	/building	JSON Object: Building	Create building



Manage Works	GET	/building/{user_id}	Empty	Get list of buildings by user
	GET	/building/{building_id}	Empty	Get building
	PUT	/building/{building_id}	JSON Object: Building	Update building
	DELETE	/building/{building_id}	Empty	Delete building
	POST	/work	JSON Object: Work	Create work
	GET	/work/{building_id}	Empty	Get list of works by building
	GET	/work/{user_id}	Empty	Get list of works by user
	GET	/work/{work_id}	Empty	Get work
	PUT	/work/{work_id}	JSON Object: Work	Update work
	PUT	/work/{work_id}/updateStatus	JSON string: status	Update work status

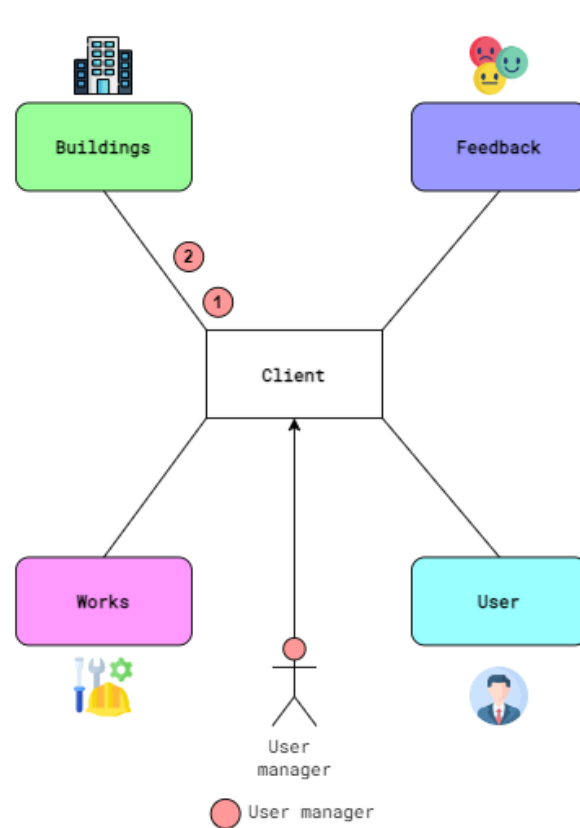
		DELETE	/work/{work_id}	Empty	Delete work
	Manage Feedback	POST	/feedback	JSON Object: Feedback	Create feedback
		GET	/feedback/{work_id}	Empty	Get list of feedbacks by work
		GET	/feedback/{feedback_id}	Empty	Get feedback
		PUT	/feedback/{feedback_id}	JSON Object: Feedback	Update feedback
		DELETE	/feedback/{feedback_id}	Empty	Delete feedback

Workflows

Building Management (Add Building)

Buildings API		
2	POST	/building
	GET	/building/{building_id} {building_details}
1	GET	/building/{user_id} {buildings_list}
	DELETE	/building/{building_id}
	PUT	/building/{building_id}

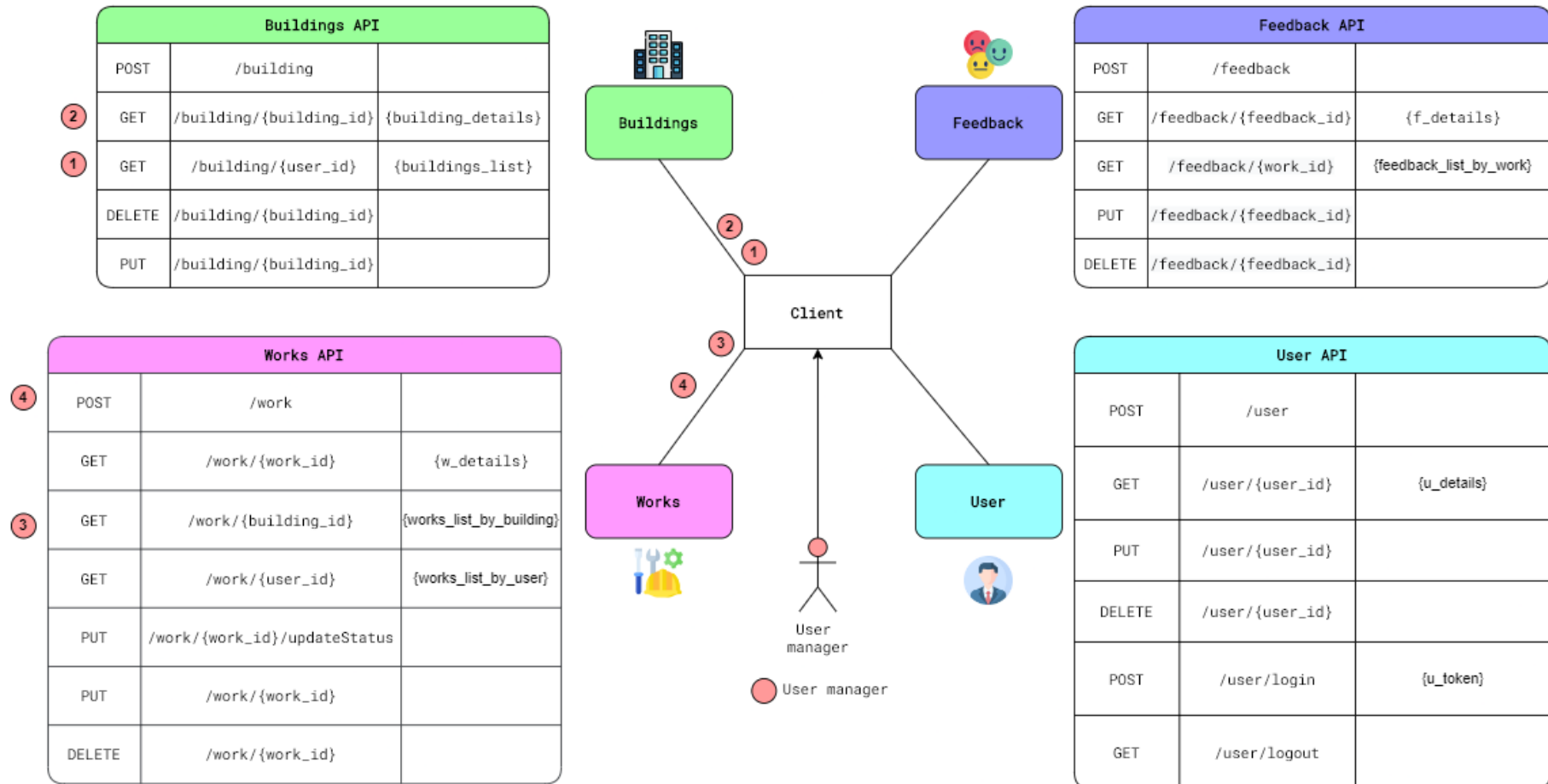
Works API		
	POST	/work
	GET	/work/{work_id} {w_details}
	GET	/work/{building_id} {works_list_by_building}
	GET	/work/{user_id} {works_list_by_user}
	PUT	/work/{work_id}/updateStatus
	PUT	/work/{work_id}
	DELETE	/work/{work_id}



Feedback API		
	POST	/feedback
	GET	/feedback/{feedback_id} {f_details}
	GET	/feedback/{work_id} {feedback_list_by_work}
	PUT	/feedback/{feedback_id}
	DELETE	/feedback/{feedback_id}

User API		
	POST	/user
	GET	/user/{user_id} {u_details}
	PUT	/user/{user_id}
	DELETE	/user/{user_id}
	POST	/user/login {u_token}
	GET	/user/logout

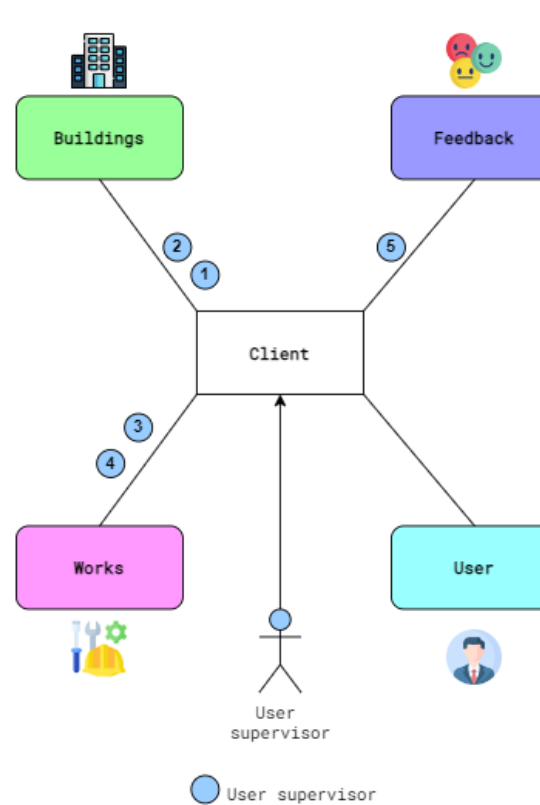
Work Management (Manager Adds Work)



Work Management (Give Feedback)

Buildings API			
	POST	/building	
2	GET	/building/{building_id}	{building_details}
1	GET	/building/{user_id}	{buildings_list}
	DELETE	/building/{building_id}	
	PUT	/building/{building_id}	

Works API			
	POST	/work	
4	GET	/work/{work_id}	{w_details}
3	GET	/work/{building_id}	{works_list_by_building}
	GET	/work/{user_id}	{works_list_by_user}
	PUT	/work/{work_id}/updateStatus	
	PUT	/work/{work_id}	
	DELETE	/work/{work_id}	



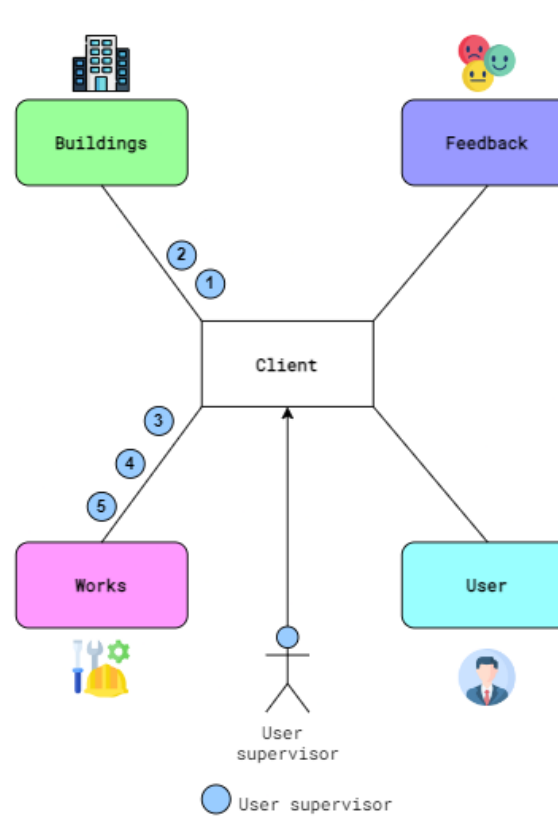
Feedback API			
	POST	/feedback	
	GET	/feedback/{feedback_id}	{f_details}
	GET	/feedback/{work_id}	{feedback_list_by_work}
	PUT	/feedback/{feedback_id}	
	DELETE	/feedback/{feedback_id}	

User API			
	POST	/user	
	GET	/user/{user_id}	{u_details}
	PUT	/user/{user_id}	
	DELETE	/user/{user_id}	
	POST	/user/login	{u_token}
	GET	/user/logout	

Work Management (Inspection)

Buildings API		
POST	/building	
2	GET	/building/{building_id} {building_details}
1	GET	/building/{user_id} {buildings_list}
	DELETE	/building/{building_id}
	PUT	/building/{building_id}

Works API		
	POST	/work
4	GET	/work/{work_id} {w_details}
3	GET	/work/{building_id} {works_list_by_building}
	GET	/work/{user_id} {works_list_by_user}
5	PUT	/work/{work_id}/updateStatus
	PUT	/work/{work_id}
	DELETE	/work/{work_id}



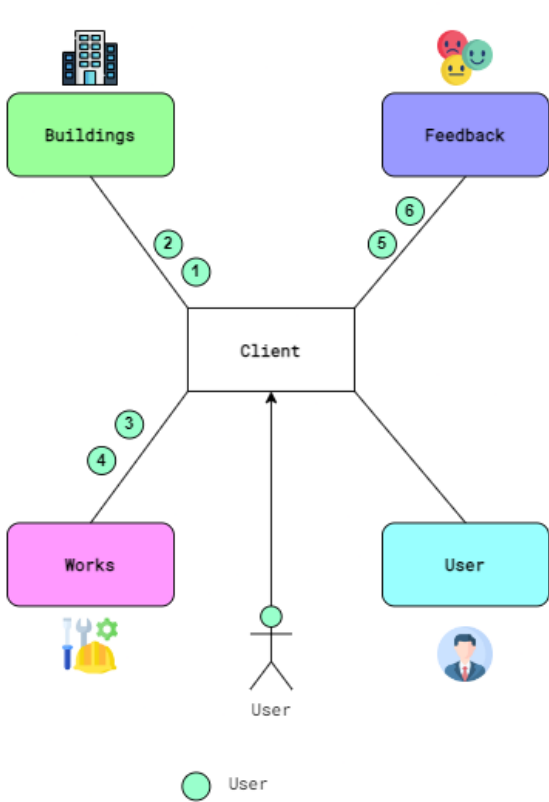
Feedback API		
POST	/feedback	
	GET	/feedback/{feedback_id} {f_details}
	GET	/feedback/{work_id} {feedback_list_by_work}
	PUT	/feedback/{feedback_id}
	DELETE	/feedback/{feedback_id}

User API		
	POST	/user
	GET	/user/{user_id} {u_details}
	PUT	/user/{user_id}
	DELETE	/user/{user_id}
	POST	/user/login {u_token}
	GET	/user/logout

Work Management (View Feedback)

Buildings API			
	POST	/building	
2	GET	/building/{building_id}	{building_details}
1	GET	/building/{user_id}	{buildings_list}
	DELETE	/building/{building_id}	
	PUT	/building/{building_id}	

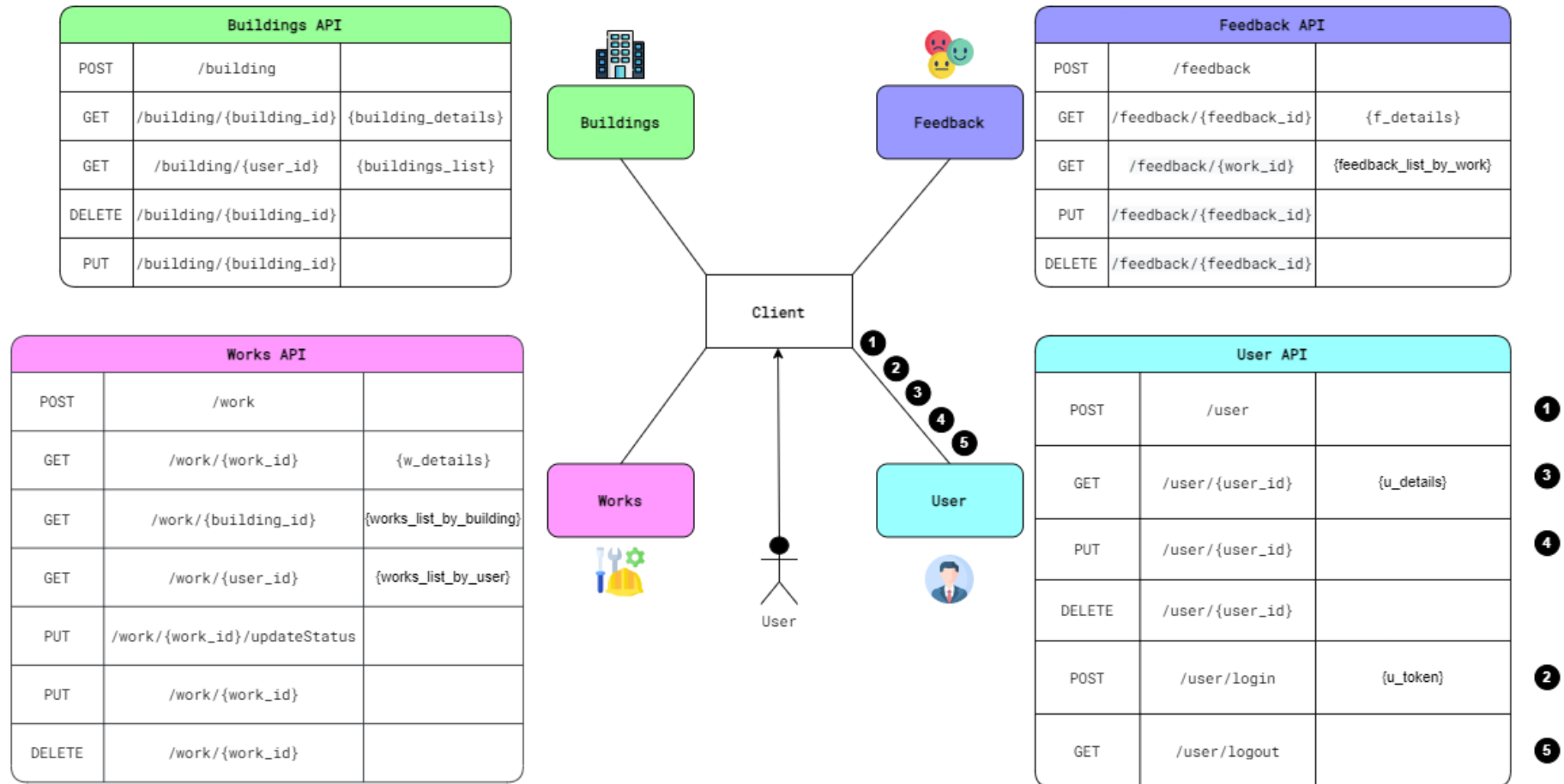
Works API			
	POST	/work	
4	GET	/work/{work_id}	{w_details}
3	GET	/work/{building_id}	{works_list_by_building}
	GET	/work/{user_id}	{works_list_by_user}
	PUT	/work/{work_id}/updateStatus	
	PUT	/work/{work_id}	
	DELETE	/work/{work_id}	



Feedback API			
	POST	/feedback	
	GET	/feedback/{feedback_id}	{f_details}
6	GET	/feedback/{work_id}	{feedback_list_by_work}
	PUT	/feedback/{feedback_id}	
5	DELETE	/feedback/{feedback_id}	

User API			
	POST	/user	
	GET	/user/{user_id}	{u_details}
	PUT	/user/{user_id}	
	DELETE	/user/{user_id}	
	POST	/user/login	{u_token}
	GET	/user/logout	

User Management (Register -> Login -> Update Profile -> Logout)



Business Objects

```

    },
    "User": {
      "type": "object",
      "required": [
        "user_id"
      ],
      "properties": {
        "user_id": {
          "type": "integer",
          "format": "int64"
        },
        "first_name": {
          "type": "string"
        },
        "last_name": {
          "type": "string"
        },
        "username": {
          "type": "string"
        },
        "email": {
          "type": "string"
        },
        "password": {
          "type": "string",
          "format": "password"
        },
        "phone_number": {
          "type": "number"
        },
        "role": {
          "type": "string",
          "enum": [
            "manager",
            "supervisor"
          ]
        }
      }
    }
  },
}

```

```

    },
    "Building": {
      "type": "object",
      "required": [
        "building_id"
      ],
      "properties": {
        "building_id": {
          "type": "integer",
          "format": "int64"
        },
        "image": {
          "type": "string",
          "format": "binary"
        },
        "name": {
          "type": "string"
        },
        "description": {
          "type": "string"
        },
        "localization": {
          "type": "string"
        },
        "problems": {
          "type": "array",
          "items": {
            "type": "object",
            "properties": {
              "description": {
                "type": "string"
              },
              "work_id": {
                "type": "integer",
                "format": "int64"
              }
            }
          }
        },
        "managers": {
          "type": "array",
          "description": "Managers IDs (assigned by system at account creation) List",
          "items": {
            "type": "integer",
            "format": "int64"
          }
        },
        "supervisors": {
          "type": "array",
          "description": "Supervisor IDs (assigned by system at account creation) List",
          "items": {
            "type": "integer",
            "format": "int64"
          }
        }
      }
    }
  },
}

```

```

    },
    "Work": {
      "type": "object",
      "required": [
        "work_id"
      ],
      "properties": {
        "work_id": {
          "type": "integer",
          "format": "int64"
        },
        "building_id": {
          "type": "integer",
          "format": "int64"
        },
        "type": {
          "type": "string"
        },
        "description": {
          "type": "string"
        },
        "startDate": {
          "type": "string",
          "format": "date"
        },
        "finishDate": {
          "type": "string",
          "format": "date"
        },
        "status": {
          "type": "string",
          "enum": [
            "closed",
            "waiting",
            "scheduled",
            "opened",
            "approved",
            "rejected"
          ]
        },
        "contractors": {
          "type": "array",
          "items": {
            "type": "object",
            "properties": {
              "name": {
                "type": "string"
              },
              "age": {
                "type": "number"
              },
              "phone_number": {
                "type": "number"
              },
              "email": {
                "type": "string"
              },
              "occupation": {
                "type": "string"
              }
            }
          }
        }
      }
    }
  },
}

```

```
"Feedback":{
  "type": "object",
  "required": [
    "feedback_id"
  ],
  "properties": {
    "feedback_id": {
      "type": "integer",
      "format": "int64"
    },
    "work_id": {
      "type": "integer",
      "format": "int64"
    },
    "user_id": {
      "type": "integer",
      "format": "int64"
    },
    "date": {
      "type": "string",
      "format": "date-time"
    },
    "title": {
      "type": "string"
    },
    "content": {
      "type": "string"
    }
  }
},
```

Data Bindings

Divs/Forms	Id	HTML	Bindings	Description
<div>Header (1)</div> <div> <div>Label</div> <div> <div>Label</div> <div>Input (2)</div> </div> <div> <div>Label</div> <div>Input (3)</div> </div> </div> <div>Button space (4,5)</div> <div>Feedback (6)</div> <div>Page - Login</div>	1	<p>	ng-bind="login"	Login title header.
	2	<input>	ng-model="username"	Input for username.
	3	<input>	ng-model="password"	Input for password.
	4	<button>	ng-click="login()"	Button to send login data to the server.
	5	<button>	ng-click="cancel()"	Button to cancel the action.
	6	<p>	ng-bind="login_feedback()"	Show feedback for login action.
<div>Header (1)</div>	1	<p>	ng-bind="create_account"	Create Account title header.

Label		2	<input>	ng-model="first_name"	Input First name
Label	Input(2)	3	<input>	ng-model="last_name"	Input Last name
Label	Input(3)	4	<input>	ng-model="username"	Input for username.
Label	Input (4)	5	<input>	ng-model="email"	Input for email.
Label	Input (5)	6	<input>	ng-model="password"	Input for password.
Label	Input (6)	7	<input>	ng-model="phone_number"	Input for phone number.
Label	Input (7)	8	<button>	ng-click="create_account()"	Button to send account data to the server.
		9	<p>	ng-bind="create_account_feedback()"	Show feedback for register action.
Button space (8)					
Feedback (9)					
Page - Create Account					

<div>Header (1)</div> <div> <div>Label</div> <div> <div>Label</div> <div>Input (2)</div> </div> <div> <div>Label</div> <div>Input (3)</div> </div> <div> <div>Label</div> <div>Input (4)</div> </div> <div> <div>Label</div> <div>Input (5)</div> </div> <div> <div>Label</div> <div>Input (6)</div> </div> </div> <div>Button space (7,8,9)</div> <div>Feedback (10)</div>	1	<p>	ng-bind="profile"	Update user's profile title header.
	2	<input>	ng-model="name"	Input for name.
	3	<input>	ng-model="username"	Input for username.
	4	<input>	ng-model="email"	Input for email.
	5	<input>	ng-model="password"	Input for password.
	6	<input>	ng-model="phone_number"	Input for phone number.
	7	<button>	ng-click="update_user_profile()"	Button to send updated profile data to the server.
	8	<button>	ng-click="cancel()"	Button to cancel the action.
	9	<button>	ng-click="delete_user_profile()"	Button to delete user from the system.

<div data-bbox="203 196 728 272">Page - Update Profile</div>	10	<p>	ng-bind="update_profile_feedback()"	Show feedback for update profile action.
<div data-bbox="203 352 728 486">Header (1)</div> <div data-bbox="203 486 728 606">Paragraph space (2,3,4,5)</div> <div data-bbox="203 606 728 1056"> <div data-bbox="203 606 728 740">Label</div> <div data-bbox="203 740 728 817">Label <div data-bbox="320 740 712 807">li (6)</div></div> <div data-bbox="203 817 728 908">Label <div data-bbox="320 817 712 908">li (7)</div></div> <div data-bbox="203 908 728 1056">Label <div data-bbox="320 908 712 1007">li (8)</div></div> </div> <div data-bbox="203 1056 728 1176">Button space (9)</div> <div data-bbox="203 1176 728 1295">Feedback (10)</div>	1	<p>	ng-bind="building_view"	Building's view title header.
	2	<p>	ng-bind="building_id"	Show building's id.
	3	<p>	ng-bind="name"	Show building's name.
	4	<p>	ng-bind="description"	Show building's description.
	5	<p>	ng-bind="localization"	Show building's localization.
	6		ng-repeat="problem in problems"	List building problems.
	7		ng-repeat="manager in managers"	List building managers.
	8		ng-repeat="supervisor in supervisors"	List building supervisors.

<div>Page - Building Overview</div>	9	<button>	ng-click="redirect_to_list_of_works()"	Button to redirect user to building works view.
	10	<p>	ng-bind="building_view_feedback()"	Show feedback for building view action.
<div>Header (1)</div> <div> <div>Label</div> <div>Label Input(2)</div> <div>Label Input (3)</div> <div>Label Input (4)</div> <div>Label Input (5)</div> <div>Label Input (6)</div> <div> <div>Label li</div> <div>Label Button(7)</div> </div> </div>	1	<p>	ng-bind="add_building"	Add building title header.
	2	<input>	ng-upload ="add_image"	Add building's image
	3	<input>	ng-model="name"	Input for name.
	4	<input>	ng-model="description"	Input for description.
	5	<input>	ng-model="localization"	Input for username.
	6	<input>	ng-model="problem" ng-click="add_problem()"	Input for problem to be added to building's problems list.
	7	<button>	ng-repeat="problem in problems" ng-click="delete_problem()"	Delete the selected problem from the building's problems list.

<div> <div></div> <div>Label</div> <div>Input (8)</div> <div>Label</div> <div>li</div> <div>Label</div> <div>Button (9)</div> <div>Label</div> <div>Input (10)</div> <div>Label</div> <div>li</div> <div>Label</div> <div>Button (11)</div> </div>	8	<input>	ng-model="manager" ng-click="add_manager()"	Input for manager to be added to building's managers list.
	9	<button>	ng-repeat="manager in managers" ng-click="delete_manager()"	Delete the selected manager from the building's managers list.
	10	<input>	ng-model="supervisor" ng-click="add_supervisor()"	Input for supervisor to be added to the building's supervisors list.
	11	<button>	ng-repeat="supervisor in supervisors" ng-click="delete_supervisor()"	Delete the selected supervisor from the building's supervisors list.
	12	<button>	ng-click="add_building()"	Button to send new building data to the server.
	13	<button>	ng-click="cancel()"	Button to cancel action.
	14	<p>	ng-bind="add_building_feedback()"	Show feedback for add building action.
Button space (12,13)				
Feedback (14)				
Page - Add Building				

		1	<p>	ng-bind="update_building"	Update building title header.
Header (1)		2	<input>	ng-model="name"	Input for name.
Label		3	<input>	ng-model="description"	Input for description.
Label	Input (2)	4	<input>	ng-model="localization"	Input for localization.
Label	Input (3)	5	<input>	ng-model="problem" ng-click="add_problem()"	Input for problem to be added to building's problems list.
Label	Input (4)	6	<button>	ng-repeat="problem in problems" ng-click="delete_problem()"	Delete the selected problem from the building's problems list.
Label	Input(5)	7	<input>	ng-model="manager" ng-click="add_manager()"	Input for manager to be added to building's managers list.
Label	li Label Button (6)	8	<button>	ng-repeat="manager in managers" ng-click="delete_manager()"	Delete the selected manager from the building's managers list.
Label	Input(7)	9	<input>	ng-model="supervisor" ng-click="add_supervisor()"	Input for supervisor to be added to the building's supervisors list.
Label	li Label Button (8)	10	<button>	ng-repeat="supervisor in supervisors" ng-click="delete_supervisor()"	Delete the selected supervisor from the building's supervisors list.

<div> <div></div> <div>Label<div>Input(9)</div></div> <div>Label<div>li<div>Label<div>Button (10)</div></div></div> </div> </div>	11	<button>	ng-click="uptade_building()"	Button to send updated building data to the server.
	12	<button>	ng-click="delete_building()"	Button to delete building from the system.
	13	<button>	ng-click="cancel()"	Button to cancel action.
	14	<p>	ng-bind="update_building_feedback()"	Show feedback for update building action.
<div>Button space (11,12,13)</div> <div>Feedback (14)</div> <div>Page - Update Building</div>				
<div>Header (1)</div> <div>Label</div>	1	<p>	ng-bind="list_buildings"	List User's buildings title header.

<div> <div>Label <input type="text" value="Input (2)"/></div> <div>Label <input type="text" value="li (3)"/></div> </div>	2	<input>	ng-change="search_building"	Search for a specific building.
	3		ng-repeat="building_in_buildings" ng-click="redirect_to_building()"	List User's building items.
	4	<button>	ng-click="redirect_to_add_building()"	Button to redirect user to add building view.
	5	<p>	ng-bind="list_building_feedback()"	Show feedback for list buildings action.
<div> <div>Header (1)</div> <div>Paragraph space (2,3,4,5,6,7)</div> <div> <div>Label</div> <div>Label <input type="text" value="Input(8)"/></div> </div> </div>	1	<p>	ng-bind="work_view"	Work's view title header.
	2	<p>	ng-bind="work_id"	Show work's id.
	3	<p>	ng-bind="building_id"	Show building's id associated with the current work.
	4	<p>	ng-bind="type"	Show work's type.
	5	<p>	ng-bind="description"	Show work's description.

<div> <div>Label<div>li (9)</div></div> <div>Button space (10,11, 12)</div> <div>Feedback (13)</div> <div>Page - Work Overview</div> </div>	6	<p>	ng-bind="startDate"	Show work's start date.
	7	<p>	ng-bind="finishDate"	Show work's finish date.
	8	<input>	ng-model="status"	Current work's status can be updated.
	9		ng-repeat="contractor in contractors"	List work contractors.
	10	<button>	ng-click="redirect_to_add_work_feedback()"	Button to redirect user to add new work's feedback page.
	11	<button>	ng-click="redirect_to_list_of_feedbacks()"	Button to redirect user to list feedbacks by building page.
	12	<button>	ng-click="update_work_status()"	Button to send updated work's status data to the server.
<div> <div>Header (1)</div> <div> <div>Label</div> <div>Label<div>Input (2)</div></div> </div> </div>	13	<p>	ng-bind="work_view_feedback()"	Show feedback for work view action.
	1	<p>	ng-bind="add_work"	Add work title header.
	2	<input>	ng-model="building_id"	Input for building's id associated with the new work.
	3	<input>	ng-model="type"	Input for type.
	4	<input>	ng-model="description"	Input for description.

<div> <div>Label</div> <div>Input (3)</div> </div> <div> <div>Label</div> <div>Input (4)</div> </div> <div> <div>Label</div> <div>Input (5)</div> </div> <div> <div>Label</div> <div>Input (6)</div> </div> <div> <div>Label</div> <div>Input (7)</div> </div> <div> <div>Label</div> <div>Input (8.1)</div> </div> <div> <div>Label</div> <div>Input (8.2)</div> </div> <div> <div>Label</div> <div>Input (8.3)</div> </div> <div> <div>Label</div> <div>Input (8.4)</div> </div> <div> <div>Label</div> <div>Input (8.5)</div> </div> <div> <div>button (8.6)</div> </div> <div> <div>Label</div> <div>li</div> <div> <div>Label</div> <div>Button (9)</div> </div> </div>	5	<input>	ng-model="startDate"	Input for start date.
	6	<input>	ng-model="finishDate"	Input for finish date.
	7	<input>	ng-model="status"	Input for status.
	8.1	<input>	ng-model="name"	Input for contractor's name.
	8.2	<input>	ng-model="age"	Input for contractor's age.
	8.3	<input>	ng-model="phone_number"	Input for contractor's phone number.
	8.4	<input>	ng-model="email"	Input for contractor's email.
	8.5	<input>	ng-model="occupation"	Input for contractor's occupation.
	8.6	<button>	ng-click="add_contractor()"	Button to add contractor to work's contractor list.
	9	<button>	ng-repeat="contractor in contractors" ng-click="delete_contractor()"	Delete the selected contractor from the work's contractors list.
	10	<button>	ng-click="add_work()"	Button to send new work data to the server.

<div> <div></div> <div>Button space (10,11)</div> <div>Feedback (12)</div> <div>Page - Add Work</div> </div>	11	<button>	ng-click="cancel()"	Button to cancel action.
	12	<p>	ng-bind="add_work_feedback()"	Show feedback for add work action.
	1	<p>	ng-bind="update_work"	Update work title header.
	2	<input>	ng-model="type"	Input for type.
<div> <div>Header (1)</div> <div> <div>Label</div> <div> <div>Label</div> <div>Input (2)</div> </div> <div> <div>Label</div> <div>Input (3)</div> </div> <div> <div>Label</div> <div>Input (4)</div> </div> </div> </div>	3	<input>	ng-model="description"	Input for description.
	4	<input>	ng-model="startDate"	Input for start date.
	5	<input>	ng-model="finishDate"	Input for finish date.
	6	<input>	ng-model="status"	Input for status date.

<div> <div>Label</div> <div>Input (5)</div> </div> <div> <div>Label</div> <div>Input (6)</div> </div> <div> <div>Label</div> <div>Input (8.1)</div> <div>Label</div> <div>Input (8.2)</div> <div>Label</div> <div>Input (8.3)</div> <div>Label</div> <div>Input (8.4)</div> <div>Label</div> <div>Input (8.5)</div> <div>button (8.6)</div> </div> <div> <div>Label</div> <div>li</div> <div> <div>Label</div> <div>Button (9)</div> </div> </div> <div>Button space (10,11,12)</div> <div>Feedback (13)</div>	8.1	<input>	ng-model="name"	Input for contractor's name.
	8.2	<input>	ng-model="age"	Input for contractor's age.
	8.3	<input>	ng-model="phone-number"	Input for contractor's phone number.
	8.4	<input>	ng-model="email"	Input for contractor's email.
	8.5	<input>	ng-model="occupation"	Input for contractor's occupation.
	8.6	<button>	ng-click="add_contractor()"	Button to add contractor to work's contractor list.
	9	<button>	ng-repeat="contractor in contractors" ng-click="delete_contractor()"	Delete the selected contractor from the work's contractors list.
	10	<button>	ng-click="update_work()"	Button to send updated work data to the server.
	11	<button>	ng-click="delete_work()"	Button to delete work from the system.
	12	<button>	ng-click="cancel()"	Button to cancel action.
	13	<p>	ng-bind="update_work_feedback()"	Show feedback for update work action.

Page - Update Work				
Header (1)	1	<p>	ng-bind="list_works_by_building"	List of works by building title header.
Label	2	<input>	ng-change="search_work"	Search for a specific work
Label Input(2)	3		ng-repeat="work in works" ng-click="redirect_to_work()"	List building's works items, when clicked redirect the user to selected work.
Label li (3)	4	<button>	ng-click="redirect_to_add_building_work"	Button to redirect user to add a building's work (microservice) view.
Button Space (4)	5	<p>	ng-bind="list_work_feedback()"	Show feedback for list works action.
Feedback (5)				
Page - Works List By Building				

<div>Header (1)</div> <div> <div>Label</div> <div> <div>Label</div> <div>Input(2)</div> </div> <div> <div>Label</div> <div>li (3)</div> </div> </div> <div>Button Space (4)</div> <div>Feedback (5)</div> <div>Page - Works List By User</div>	1	<p>	ng-bind="list_works_by_user"	List of works by user title header.
	2	<input>	ng-change="search_work"	Search for a specific work
	3		ng-repeat="work in works" ng-click="redirect_to_work()"	List user's works items, when clicked redirect the user to selected work.
	4	<button>	ng-click="redirect_to_add_work"	Button to redirect user to add a building's work (microservice) view.

	5	<p>	ng-bind="list_work_feedback()"	Show feedback for list works action.
<div>Header (1)</div> <div>Paragraph space (2,3,4,5,6,7)</div> <div>Button Space (8)</div> <div>Feedback (9)</div> <div>Page - Feedback Overview</div>	1	<p>	ng-bind="feedback_view"	Work's Feedback Message view title header.
	2	<p>	ng-bind="feedback_id"	Show work's feedback message id.
	3	<p>	ng-bind="work_id"	Show work's id associated with the current work's feedback message.
	4	<p>	ng-bind="username"	Show the username of the work's feedback message author.
	5	<p>	ng-bind="date"	Show work's feedback message date.
	6	<p>	ng-bind="title"	Show work's feedback message title.
	7	<p>	ng-bind="content"	Show work's feedback message content.
	8	<button>	ng-click="redirect_to_update_feedback"	Button to redirect user to update work feedback message.

	9	<p>	ng-bind="feedbackMessage_view_feedback()"	Show feedback for work's feedback message view action.
Header (1)	1	<p>	ng-bind="add_feedback"	Add work's feedback message title header.
Label	2	<input>	ng-model="work_id"	Input for work's id associated with the current work's feedback message .
Label Input (2)	3	<input>	ng-model="title"	Input for title.
Label Input (3)	4	<textarea>	ng-model="content"	Input for content.
Label Textarea (4)	5	<button>	ng-click="add_feedback()"	Button to send new work's feedback message data to the server (datetime and author's info, ie the current user, don't need user's input).
Button space (5,6)	6	<button>	ng-click="cancel()"	Button to cancel action.
Feedback (7)	7	<p>	ng-bind="add_feedbackMessage_feedback()"	Show feedback for add work's feedback message action.
Page - Add Feedback				

<div>Header (1)</div> <div> <div>Label</div> <div> <div>Label</div> <div>Textarea (2)</div> </div> </div> <div>Button space (3,4,5)</div> <div>Feedback (6)</div> <div>Page - Update Feedback</div>	1	<p>	ng-bind="update_feedback"	Update work's feedback message title header.
	2	<textarea>	ng-model="content"	Input for content.
	3	<button>	ng-click="update_feedback()"	Button to send updated work's feedback message data to the server.
	4	<button>	ng-click="cancel()"	Button to cancel action.
	5	<button>	ng-click="delete_feedback()"	Button to delete work's feedback message from the system.
	7	<p>	ng-bind="update_feedbackMessage_feedback()"	Show feedback for update work's feedback message action.
<div>Header (1)</div>	1	<p>	ng-bind="list_feedbacks"	List work's feedback message title header.

<div>Label</div> <div>Label Input(2)</div> <div>Label li (3)</div>	2	<input>	ng-change="search_feedback"	Search for a specific feedback
	3		ng-repeat="feedback in feedbacks" ng-click="redirect_to_feedback()"	List work's feedback items, when clicked redirect the user to selected work's feedback message .
	4	<p>	ng-bind="list_feedbackMessages_feedback()"	Show feedback for list work's feedback messages action.
Page - Feedbacks				
<div>Button Space (1,2,3,4)</div> <div>Header (present on all pages except login and account creation pages)</div>	1	<button>	ng-click="redirect_to_buildings"	Button to redirect user to buildings list view.
	2	<button>	ng-click="redirect_to_works_list_by_user()"	Button to redirect user to works list view.

	3	<button>	ng-click="redirect_to_add_feedback()"	Button to redirect user add feedback form.
	4	<button>	ng-click="redirect_to_user_profile()"	Button to redirect user to profile view.

Bootstrap




Add Building

Upload Building Image

Choose File

No file chosen

 Name

 Description

 Localization

Problem



Added problems:

⊙ Problem #1



Manager's username



Added Managers:

⊙ manager_username01



Supervisor's username



Added Supervisors:

⊙ supervisor_username01



Submit

Cancel



Send Feedback



Send

Cancel

or call us via email




Add Work

 Type


 Description

Start Date

 mm/dd/yyyy



Finish Date

 mm/dd/yyyy



Status

Closed



Add Contractor:

 Name

 Age

 Phone Number

 Email

 Occupation

+


Added Contractors:

• Contractor 1





Submit

Cancel



BuildingsWorksFeedback





Building ID:

123

Building Name:

Example Building Name

Localization:

Lisboa, Portugal

Description:

Building's description.

Problems:

Problem #1

Problem #1

Problem #1

Problem #1

View Building's Works

Managers:

Manager Name 1

Manager Name 2

Manager Name 3

Manager Name 4


Supervisors:

Supervisor Name 1


Supervisor Name 2

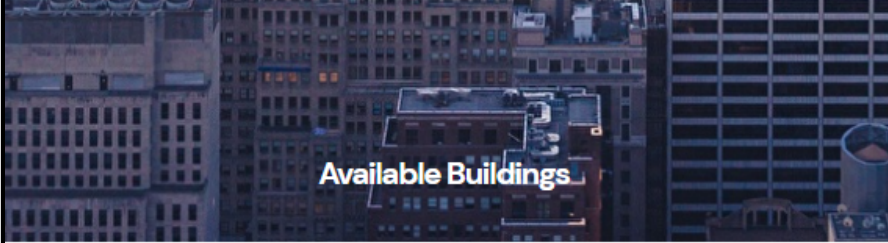
Supervisor Name 3

Supervisor Name 4




BuildingsWorksFeedback






Search

Add Building




Lisboa, Portugal

Building Name 1




Porto, Portugal

Building Name 2




Setúbal, Portugal

Building Name 3



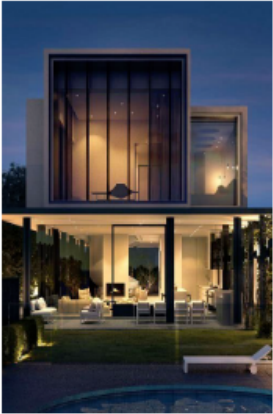
Faro, Portugal

Building Name 4



Braga, Portugal

Building Name 5



Evora, Portugal

Building Name 6



Create Account

Sign Up


Already have an account? [Log In](#)




Feedback

Feedback ID:	2
Work ID:	1
Username:	username
Date:	08-06-2022 · 3h01pm
Title:	Feedback Title
Content: Feedback message content.	


[Update Feedback](#)


 BUILDING
MAINTENANCE APP


BuildingsWorksFeedback





Feedbacks


 **Feedback #1**
Author: Username
Date: 08-06-2022 · 3h01 pm


 **Feedback #2**
Author: Username
Date: 08-06-2022 · 3h02 pm

 **Feedback #3**
Author: Username
Date: 08-06-2022 · 3h03 pm

 **Feedback #4**
Author: Username
Date: 08-06-2022 · 3h04 pm

 **Feedback #5**
Author: Username
Date: 08-06-2022 · 3h05 pm

 **Feedback #6**
Author: Username
Date: 08-06-2022 · 3h06 pm



Sign In

Sign In

Don't have an account? [Create Account](#)



Profile

Username

Name

Email

Phone Number

New Password

Confirm New
Password

Update Profile

Cancel

Delete



Update Building

Upload Building Image

Choose File

No file chosen

Name

Description

Localization

Problem



Added problems:

✖ Problem #1



Manager's username



Added Managers:

✖ manager_username01



Supervisor's username



Added Supervisors:

✖ supervisor_username01



Submit

Cancel

Delete



Update Feedback

Your message




Update

Cancel

Delete




Update Work

 Type

 Description

Start Date

 mm/dd/yyyy



Finish Date

 mm/dd/yyyy




Status


Closed



Add Contractor:

 Name

 Age

 Phone Number

 Email

 Occupation

+

Added Contractors:

✖ Contractor 1



Submit

Cancel

Delete



Work

Work ID:

1

Building ID:

123

Type:

Painting

Description:

Work's description.

Start Date:

03-05-2022

Finish Date:

10-05-2022

Status

Closed



Update Status

Contractors:

✱ Contractor Name 1

✱ Contractor Name 2

✱ Contractor Name 3

✱ Contractor Name 4

Send Feedback

Feedbacks



Works

[Add Work](#)

Work #1

Work #1 type.



Work #2

Work #2 type.



Work #3

Work #3 type.



Work #4

Work #4 type.



Work #5

Work #5 type.



Work #6

Work #6 type.

Story Spaces

App Development Canvas



Building Maintenance App

Concept

Aplicação para monitorização de trabalhos de construção e manutenção de edifícios.



Problem

Which problem are you solving and why is it urgent



Não existência de ferramentas que forneçam os níveis de agilidade e flexibilidade necessários à equipa BIT que apenas está focada na inspeção e de ter os trabalhos concluídos.

Target group

The audience you aim to target



Equipas de inspeção e supervisão de trabalhos em edifícios (BIT).

Discards

Which apps or solutions will users leave behind for yours



Outras ferramentas de gestão de projetos, que não dão tanta agilidade e flexibilidade como esta.

Deixa de ser necessário à BIT, estar constantemente a inspecionar e supervisionar os trabalhos de forma presencial.

Characteristics

Identifying features of the app



- Listagem dos edifícios com a suas informações base, informações sobre os managers e supervisores dos trabalhos de manutenção a serem feitos em cada edifício e problemas a serem resolvidos.
- Listagem dos trabalhos específicos a serem realizados num determinado edifício com informações importantes como datas, descrições, estado e contratados.
- Espaços para adição, edição e remoção de edifícios e/ou trabalhos de manutenção.
- Espaço para ver e dar feedback sobre o trabalho.

Success factors

Factors that will help you succeed



- Inexistência de ferramentas com o nível de simplicidade desejado.
- Pouca concorrência para o público alvo marcado.
- Pouco custo financeiro, temporal e humano para a realização da aplicação.

Core functions

Core features of your app and their prioritization



- Serviço de edifícios.
- Serviço de trabalhos de manutenção.
- Serviço de gestão de utilizadores.
- Serviço de feedback.

Platforms

REST API, NoSQL, Mobile, Web



Languages

Português e Inglês



Vision

What would you like to accomplish within a certain timeframe



Uma grande afluência à app por parte da BIT ao fim de um ano.

Diagrams

