

# MetroCar

## Plán testování

Autor: Dominik Moštěk <mostedom@fel.cvut.cz>  
Ostatní členové: Jakub Ječmínek  
Alexandr Makarič  
Martin Kinčl

Datum: 19.12.2011  
Předmět: A7B36SI2

## Revize

Verze	Datum	Autor	Změny
0.1	19.12.2011	Dominik Moštěk	Sestavení počátečního plánu

## 1 Účel dokumentu

Tento dokument popisuje strategie, procesy a metody určené k plánování a provádění testů a zpracování jejich výsledků.

### 1.1 Cíle testování

Primárním cílem testování je ověření, že aplikace splňují všechny požadavky na ně kladené a to funkční i obecné. A splňují metriky stanovené v požadavcích. Cílem je též ověřit scénáře stanovené pomocí Use Case. Sekundárním cílem testování je odhalit nedostatky v návrhu aplikace.

### 1.2 Priority

Vzhledem k povaze projektu je nutné, aby následující seznam funkcionalit byl plně otestován a schopen plného provozu.

- Vytvoření rezervace skrze webové rozhraní
- Platba za služby skrze webové rozhraní.
- Odemknutí automobilu pomocí klientské aplikace
- Záznam ujetých kilometrů
- Sledování polohy automobilu

### 1.3 Role a odpovědnosti

### 1.4 Správce projektu

Správce projektu je odpovědný za provedení testování popsaného v tomto dokumentu a zpracování výsledků. Jeho odpovědností je též dohled nad výsledky oprav chyb a nedostatků ohlacených při testování.

### 1.5 Vývojář webového rozhraní

Je odpovědný za přípravu serverové části k testování. To zahrnuje poskytnutí rozhraní skrze, které se bude aplikace testovat a implementace všech nezbytných částí pro supštění testů. Jeho dalším úkolem je zanesení oprav chyb stanovených vedoucím do zdrojového kódu.

### 1.6 Vývojář mobilních aplikací

Je odpovědný za přípravu mobilních aplikací k testování a to jak k automatickému, tak testování v terénu. To zahrnuje poskytnutí rozhraní skrze, které se bude aplikace testovat a implementace všech nezbytných částí pro supštění testů. Jeho dalším úkolem je zanesení oprav chyb stanovených vedoucím do zdrojového kódu.

## 2 Testování

### 2.1 Metody testování

#### 2.1.1 Testování použitelnosti

Účelem testování použitelnosti je ujištění, zda komponenty a funkce aplikace odpovídají požadavkům zadavatele. Toto testování bude provedeno uživateli, kteří budou budoucími uživateli těchto aplikací. Podmínkou relevantnosti tohoto testu je, aby tito uživatelé nebyli součástí týmu, který aplikaci vypracovává. Testeři též musí být alespoň začátečníci v používání chytrých mobilních aplikací s OS Android. Testy použitelnosti budou prováděny na následujících částech projektu:

- Klientská aplikace
- Webové rozhraní

#### 2.1.2 Unit testy

Jednotkové testování je prováděno za účelem zjištění základní funkčnosti základních částí aplikace (jednotek). Je na odpovědnosti vedoucího projektu, jaké zvolí pokrytí kódu testy. Jednotkové testy budou prováděny na následujících jednotkách:

- Komunikační jednotka  
Test otestuje komunikaci serveru a mobilních aplikací. A to z pohledu serveru, tak i z pohledu obou mobilních aplikací.
- Dtabázové rozhraní Testy komunikace serveru s databází. Test musí obsahovat testy konzistence dat.
- Test komunikace klientské aplikace a palubní jednotky Test pokryje především odemykání a zamykání automobilu, při rezervaci a bez rezervace.

#### 2.1.3 Iterační/Regresní testování

Při každém přírůstku funkcionality je potřeba otestovat, zda předchozí celky stále fungují a nejsou novou funkcionalitou zasaženy. K tomu slouží regresní testování. Skládá se z provedení veškerých testů a sepsání zprávy o testování.

#### 2.1.4 Akceptační testy

Provedou se na základě stanovených akceptačních kritérií (stanovených v POS).

#### 2.1.5 Testy stability

Budou provedeny zátěžovými testy, tedy pokusem o zpracování velkého počtu požadavků najednou.

## 2.2 Úrovně testování

### 2.2.1 Build testy

**Level 1 - Build acceptance test** Tento test prošel, pokud je možné aplikaci sestavit. V případě webové části deploynout na server. Pokud se testování zastaví na tomto levelu je aplikace vrácena vývojáři bez provedení dalších testů.

**Level 2 - Smoke testy** Jedná se o rychlé otestování, zda je aplikace schopna spuštění a základní práce s ní. Tento test slouží jako ujištění se, zda má cenu provádět další nákladné testy. Pokud se testování zastaví na tomto levelu je aplikace vrácena vývojáři bez provedení dalších testů.

**Bug-Regression testy** Otestuje se každý bug nahlášený po minulé iteraci, který byl označen jako "Fixed" nebo "Test".

### 2.2.2 Milestone testy

Tyto testy budou provedeny minimálně jedenkrát, ale nejlépe vícekrát během jedné iterace.

**Unit testy** Proběhnou dle popisu výše.

**Kritická cesta** Proběhnou testy funkcionalit, které uživatel aplikací uvidí a přijde s nimi do běžného styku. To se jedná hlavně o testy GUI, stability a použitelnosti.

### 2.2.3 Release test

Tento test zahrnuje všechny předešlé testy. Zároveň se na této úrovni testuje instalce a správa aplikace.

## 3 Testovací software

Pro účely testování bude použito následujícího software.

- JUnit 4, Django's unittest Jednotkové testování
- Selenium Testy GUI
- JMeter Testy stability

## 4 Hlášení chyb

Po každé iteraci bude vytvořena zpráva o testování, která shrne veškeré výsledky testování. Během testování budou všechny chyby zaznamenávány do systému assembla. Každá člen vývojářského týmu je povinen nalezené chyby do tohoto systému zadávat a také nově zadané chyby sledovat a pokud mu jsou přiděleny, tak zajistit jejich opravu. Adresa ticketovacího systému assembla: [http://www.assembla.com/spaces/wagnejan\\_metrocar/tickets](http://www.assembla.com/spaces/wagnejan_metrocar/tickets)